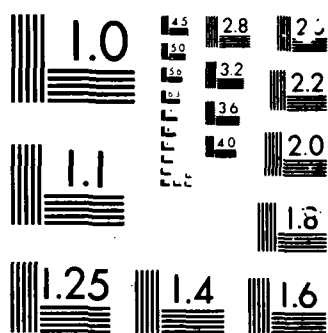


AD-A176 281 MODIFICATION AND IMPROVEMENT OF SOFTWARE FOR MODELING 1/1
MULTIDIMENSIONAL RE. (U) SCIENTIFIC COMPUTING
ASSOCIATES INC NEW HAVEN CT* M ANGEVINE ET AL NOV 86
UNCLASSIFIED AFMAL-TR-86-2897 F33615-85-C-2572 F/G 21/4 NL

/NI
2. 27
100



2

AFWAL-TR-86-2097

MODIFICATION AND IMPROVEMENT OF SOFTWARE FOR
MODELING MULTIDIMENSIONAL REACTING FUEL FLOW

Scientific Computing Associates, Inc.
246 Church St., Suite 307
New Haven, CT 06510



DTIC
ELECTE
JAN 29 1987
S D

November 1986

Final Report for Period July 1985 - February 1986

Approved for public release; distribution is unlimited.

AD-A176 281

DTIC FILE COPY

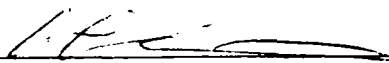
AERO PROPULSION LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6563

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

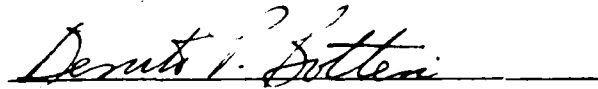


ALAN LESMERISES, 2Lt, Project Engineer
Fuels Branch
Fuels and Lubrication Division
Aero Propulsion Laboratory



ARTHUR V. CHURCHILL, Chief
Fuels Branch
Fuels and Lubrication Division
Aero Propulsion Laboratory

FOR THE COMMANDER



BENITO P. BOTTERI, Assistant Chief
Fuels and Lubrication Division
Aero Propulsion Laboratory

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFWAL/POSF, W-PAFB, OH 45433 to help us maintain a current mailing list".

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

AD-A176281

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b RESTRICTIVE MARKINGS		
2a SECURITY CLASSIFICATION AUTHORITY			3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.		
2b DECLASSIFICATION/DOWNGRADING SCHEDULE			5 MONITORING ORGANIZATION REPORT NUMBER(S) AFWAL-TR-86-2097		
4 PERFORMING ORGANIZATION REPORT NUMBER(S)			7a NAME OF MONITORING ORGANIZATION Aero Propulsion Laboratory (AFWAL/POSF) Air Force Wright Aeronautical Laboratories		
6a NAME OF PERFORMING ORGANIZATION Scientific Computing Associates, Inc.		6b OFFICE SYMBOL (If applicable)	7b ADDRESS (City, State, and ZIP Code) Wright-Patterson Air Force Base, Ohio 45433-6563		
6c ADDRESS (City, State, and ZIP Code) 246 Church St., Suite 307 New Haven, CT 06510		9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F33615-85-C-2572			
8a NAME OF FUNDING/SPONSORING ORGANIZATION		8b OFFICE SYMBOL (If applicable) AFWAL/POSF	10 SOURCE OF FUNDING NUMBERS		
8c ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB, OH 45433-6563		PROGRAM ELEMENT NO 65502F	PROJECT NO 3005	TASK NO 20	WORK UNIT ACCESSION NO. 39
11 TITLE (Include Security Classification) Modification and Improvement of Software for Modeling Multidimensional Reacting Fuel Flow					
12 PERSONAL AUTHOR(S) Mark Angevine, Mitchell Smooke and David Keyes					
13a TYPE OF REPORT Final		13b TIME COVERED FROM 85Jul15 TO 86Feb13		14 DATE OF REPORT (Year, Month, Day) 1986 November	
15 PAGE COUNT 47					
16 SUPPLEMENTARY NOTATION This is a Small Business Innovation Research Program Report.					
17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB-GROUP	Teach Code; Reacting Flows; Chemical Kinetics; Finite Rate Chemistry; Kinetic Theory Transport Effects; Flame Sheet Models; Combustion Modeling; Computational Fluid Dynamics.		
21	02				
12	01				
19 ABSTRACT (Continue on reverse if necessary and identify by block number) This report describes a project to develop computational models of reacting flows that combine the effects of detailed chemical kinetics with complex fluid mechanics to form an integrated analytical code. Initial efforts were aimed at solutions for counterflow diffusion flames; this model incorporated both finite rate chemistry and complex kinetic theory transport effects. Next, a flame sheet model was used to provide improved starting estimates for the calculations. A combination of time integration and Newton's method was used to solve the flame sheet and finite rate equations. Success with the one-dimensional model led to the development of a more generalized two-dimensional model. A parallel effort investigated ways to improve the computational efficiency of the primitive variable implementation of the TEACH code. Further generalization of these ideas to axisymmetric (laminar and turbulent) diffusion flame studies will be the focus of a follow-on project.					
20 DISTRIBUTION/AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS			21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED		
22a NAME OF RESPONSIBLE INDIVIDUAL 2Lt A.L. Lesmerises			22b TELEPHONE (Include Area Code) (513) 255-6250		22c OFFICE SYMBOL AFWAL/POSF

DD FORM 1473, 84 MAR

81 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

1. SUMMARY

The ability to predict the coupled effects of complex transport phenomena with detailed chemical kinetics is critical in an understanding of turbulent reacting flows, in improving engine efficiency, and in the study of pollutant formation. Since three-dimensional models combining both fluid dynamical effects with finite rate chemistry are as yet computationally infeasible, the modeling of chemically reacting flows generally proceeds along two independent paths. In one case chemistry is given priority over fluid mechanical effects and these models are used to assess the important elementary reaction paths, for example, in hydrocarbon fuels. In the other case, multidimensional fluid dynamical effects are emphasized with chemistry receiving little or no priority.

The goal in reacting flow computations is to be able to combine the effects of detailed chemistry with complex fluid mechanics. During the past six months, our Phase I SBIR concentrated in this direction. We investigated several areas in which the modeling of chemically reacting flows could be improved:

1. We incorporated detailed kinetics with complex transport in the solution of one-dimensional diffusion flames.
2. We investigated flame sheet initialization procedures.
3. We considered more implicit solution algorithms for the highly nonlinear discrete equations.

As a result of the limited computer resources available under Phase I combined with the computational complexity of calculating chemically reacting flows, we devoted a large part of our effort to the solution of counterflow diffusion flames. Unlike models in which the chemistry and transport phenomena are approximated by simplified relations, we incorporated both detailed finite rate chemistry and complex kinetic theory transport effects in our model. We then investigated the use of a flame sheet model as a means of providing improved starting estimates for the finite rate counterflow problem. Solution of the flame sheet and finite rate equations was accomplished with a combination of time integration and Newton's method. Based upon the success of our study in one dimension, we directed our efforts to two-dimensional problems. In particular, we generalized our one-dimensional flame sheet model to two dimensions and, in parallel with this effort, we investigated ways of improving the computational efficiency of the primitive variable implementation of the *TEACH* code. The results of our Phase I research clearly indicate the feasibility of our methodology in solving reacting systems with detailed transport and complex chemistry. This methodology has potential applications to problems in turbulent reacting flows, engine efficiency, commercial power generation units, and pollutant formation.



Dist	Special
A-1	

2. COUNTERFLOW DIFFUSION FLAMES

The flame type of most practical combustion devices is the diffusion flame. These flames are important in the interaction of heat and mass transfer with chemical reactions in ram jets, jet turbines and commercial burners. Three-dimensional models that couple the effects of fluid flow with detailed chemical reactions are as yet computationally infeasible. We can, however, obtain important information in practical systems by considering two-dimensional configurations. The counterflow diffusion flame is one such configuration.

Experimentally these flames can be produced when a reaction zone is stabilized near the stagnation point of two infinitely wide coaxial concentric jets [1] (see Figure 1). Fuel is emitted from one jet and oxidizer (air) from the other. Combustion occurs within a thin flame zone near the stagnation point where the fuel and the oxidizer are in stoichiometric proportion. Although the flow in the double jet experimental configuration is two-dimensional, the mathematical model can be reduced to the solution of a system of coupled nonlinear two-point boundary value problems along the stagnation point streamline. In this way we can investigate the effect of detailed chemical kinetics with complex transport while still having a computationally feasible problem.

2.1 Problem Formulation

Our model for counterflow diffusion flames assumes the flow to be laminar, stagnation point flow. Hence, the governing boundary layer equations for mass, momentum, chemical species and energy can be written in the form

$$\frac{\partial(\rho u x^\alpha)}{\partial x} + \frac{\partial(\rho v x^\alpha)}{\partial y} = 0, \quad (2.1)$$

$$\rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} + \frac{\partial p}{\partial x} = \frac{\partial}{\partial y} \left(\mu \frac{\partial u}{\partial y} \right), \quad (2.2)$$

$$\rho u \frac{\partial Y_k}{\partial x} + \rho v \frac{\partial Y_k}{\partial y} + \frac{\partial}{\partial y} (\rho Y_k V_{ky}) - \dot{w}_k W_k = 0, \quad k = 1, 2, \dots, K, \quad (2.3)$$

$$\rho u c_p \frac{\partial T}{\partial x} + \rho v c_p \frac{\partial T}{\partial y} - \frac{\partial}{\partial y} \left(\lambda \frac{\partial T}{\partial y} \right) + \sum_{k=1}^K \rho Y_k V_{ky} c_{pk} \frac{\partial T}{\partial y} + \sum_{k=1}^K \dot{w}_k W_k h_k = 0, \quad (2.4)$$

where α represents a geometric factor ($\alpha = 0$ for cartesian coordinates and $\alpha = 1$ for cylindrical coordinates). The system is closed with the ideal gas law,

$$\rho = \frac{p \bar{W}}{RT}. \quad (2.5)$$

In these equations x and y denote independent spatial coordinates in the tangential and transverse directions, respectively; T , the temperature; Y_k , the mass fraction of the

k^{th} species; p , the pressure; u and v the tangential and the transverse components of the velocity, respectively; ρ , the mass density; W_k , the molecular weight of the k^{th} species; \bar{W} , the mean molecular weight of the mixture; R , the universal gas constant; λ , the thermal conductivity of the mixture; c_p , the constant pressure heat capacity of the mixture; c_{pk} , the constant pressure heat capacity of the k^{th} species; \dot{w}_k , the molar rate of production of the k^{th} species per unit volume; h_k , the specific enthalpy of the k^{th} species; μ the viscosity of the mixture and V_{ky} is the diffusion velocity of the k^{th} species in the y direction. In both configurations the free stream (tangential) velocity at the edge of the boundary layer is given by $u_\infty = ax$ where a is the strain rate.

Upon introducing the notation

$$f' = \frac{u}{u_\infty}, \quad (2.6)$$

$$V = \rho v, \quad (2.7)$$

the boundary layer equations can be transformed into a system of ordinary differential equations valid along the stagnation-point streamline $x = 0$. For a system in rectangular or cylindrical coordinates, we have

$$\frac{dV}{dy} + a(1 + \alpha)\rho f' = 0, \quad (2.8)$$

$$\frac{d}{dy} \left(\mu \frac{df'}{dy} \right) - V \frac{df'}{dy} + a(\rho_\infty - \rho(f')^2) = 0, \quad (2.9)$$

$$-\frac{d}{dy}(\rho Y_k V_k) - V \frac{dY_k}{dy} + \dot{w}_k W_k = 0, \quad k = 1, 2, \dots, K, \quad (2.10)$$

$$\frac{d}{dy} \left(\lambda \frac{dT}{dy} \right) - c_p V \frac{dT}{dy} - \sum_{k=1}^K \rho Y_k V_k c_{pk} \frac{dT}{dy} - \sum_{k=1}^K \dot{w}_k W_k h_k = 0. \quad (2.11)$$

The boundary conditions for the double-jet configuration at $y = -\infty$ are given by

$$V = V_{-\infty}, \quad (2.12)$$

$$f' = \sqrt{\frac{\rho_\infty}{\rho_{-\infty}}}, \quad (2.13)$$

$$Y_k = Y_{k-\infty}, \quad k = 1, 2, \dots, K, \quad (2.14)$$

$$T = T_{-\infty}, \quad (2.15)$$

and at $y = \infty$ by

$$f' = 1, \quad (2.16)$$

$$Y_k = Y_{k\infty}, \quad k = 1, 2, \dots, K, \quad (2.17)$$

$$T = T_{\infty}. \quad (2.18)$$

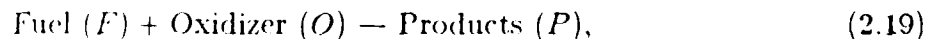
The mass flux, temperature, and species mass fractions ($V_{-\infty}, T_{-\infty}, Y_{k-\infty}$) at the fuel jet are specified quantities as are the temperature and species mass fractions (T_{∞} and $Y_{k\infty}$) at the oxidizer jet.

2.2 Flame Sheet Model

Despite the outwardly simple form of the counterflow equations, the determination of a "good" initial solution estimate can be difficult. The difficulty is due to the exponential dependence of the chemistry terms on the temperature and to the nonlinear coupling of the fluid and the thermochemistry solution fields. In previous work, we focused our efforts on the solution of adiabatic and nonadiabatic premixed laminar flames by adaptive finite difference methods [2-3]. In these problems cubic polynomials and Gaussian shaped profiles were used as starting estimates for the major and minor species on an initial coarse grid. These approximations were often sufficient to bring the starting estimates into the domain of convergence of Newton's method.

In adiabatic and nonadiabatic premixed laminar flame problems the conservation of mass and momentum reduces to the specification of a constant mass flow rate and a constant thermodynamic pressure [2-3]. Hence, thermochemical considerations play a more important role in these problems than do fluid dynamical aspects. This is not the case in counterflow diffusion flames. In particular, there is a strong coupling between the fluid dynamic and the thermochemistry solution fields in these flames. We have found that, although the solution procedure used in premixed laminar flame problems can work in selected counterflow cases, it does not provide a sufficiently robust or efficient starting estimate from which Newton's method will converge. In addition, the relaxation to steady-state (or at least until the solution is within the convergence domain of Newton's method) is very slow. The importance of these flames in turbulence modeling and in the determination of chemically controlled extinction limits, however, necessitates the development of an efficient starting procedure. We couple the appropriate equations of mass and momentum with a Shvab-Zeldovich equation to provide flame sheet starting estimates for the mass flux in the transverse direction, the similarity function, the temperature, and the stable major species in the flame.

Our starting point is the assumption that the fuel and the oxidizer obey a single overall irreversible reaction of the type



in the presence of an inert gas (N). We have

$$\nu_F F + \nu_O O \rightarrow \nu_P P, \quad (2.20)$$

where ν_F, ν_O and ν_P are the stoichiometric coefficients of the fuel, the oxidizer and the product, respectively. In addition, we neglect thermal diffusion and assume that the ordinary mass diffusion velocities can be written in terms of Fick's law, that is

$$V_k = -\frac{D_k}{Y_k} \frac{dY_k}{dy}, \quad k = 1, 2, \dots, K, \quad (2.21)$$

where D_k is the diffusion coefficient of the k^{th} species with respect to the mixture. We also take $c_p = c_{p_k}$ to be constant. With these approximations we can write

$$\frac{dV}{dy} + a(1 + \alpha)\rho f' = 0, \quad (2.22)$$

$$\frac{d}{dy} \left(\mu \frac{df'}{dy} \right) - V \frac{df'}{dy} + a(\rho_\infty - \rho(f')^2) = 0, \quad (2.23)$$

$$\frac{d}{dy} \left(\rho D_F \frac{dY_F}{dy} \right) - V \frac{dY_F}{dy} - W_F \nu_F \dot{w} = 0, \quad (2.24)$$

$$\frac{d}{dy} \left(\rho D_O \frac{dY_O}{dy} \right) - V \frac{dY_O}{dy} - W_O \nu_O \dot{w} = 0, \quad (2.25)$$

$$\frac{d}{dy} \left(\rho D_P \frac{dY_P}{dy} \right) - V \frac{dY_P}{dy} + W_P \nu_P \dot{w} = 0, \quad (2.26)$$

$$\frac{d}{dy} \left(\rho D_N \frac{dY_N}{dy} \right) - V \frac{dY_N}{dy} = 0, \quad (2.27)$$

$$\frac{d}{dy} \left(\frac{\lambda}{c_p} \frac{dT}{dy} \right) - V \frac{dT}{dy} + \frac{(W_F \nu_F h_F + W_O \nu_O h_O - W_P \nu_P h_P)}{c_p} \dot{w} = 0, \quad (2.28)$$

where

$$\dot{w} = -\frac{\dot{w}_F}{\nu_F} = -\frac{\dot{w}_O}{\nu_O} = \frac{\dot{w}_P}{\nu_P}, \quad (2.29)$$

is the rate of progress of the reaction and where we have made use of the fact that $\sum_{k=1}^K Y_k V_k = 0$.

If we introduce the heat release per unit mass of the fuel Q where

$$Q = h_F + \frac{W_O \nu_O}{W_F \nu_F} h_O - \frac{W_P \nu_P}{W_F \nu_F} h_P, \quad (2.30)$$

and if we assume that the Lewis numbers

$$\text{Le}_F = \frac{\lambda}{\rho D_F c_p}, \quad \text{Le}_O = \frac{\lambda}{\rho D_O c_p}, \quad (2.31a)$$

$$\text{Le}_P = \frac{\lambda}{\rho D_P c_p}, \quad \text{Le}_N = \frac{\lambda}{\rho D_N c_p}, \quad (2.31b)$$

are all equal to one, then each of the Shvab-Zeldovich variables

$$Z_F = Y_F - Y_{F\infty} + \frac{c_p}{Q}(T - T_\infty), \quad (2.32)$$

$$Z_O = Y_O - Y_{O\infty} + \frac{c_p W_O \nu_O}{Q W_F \nu_F}(T - T_\infty), \quad (2.33)$$

$$Z_P = Y_P - Y_{P\infty} - \frac{c_p W_P \nu_P}{Q W_F \nu_F}(T - T_\infty), \quad (2.34)$$

$$Z_N = Y_N - Y_{N\infty}, \quad (2.35)$$

satisfies the following differential equation

$$\frac{d}{dy} \left(\rho D_k \frac{dZ_k}{dy} \right) - V \frac{dZ_k}{dy} = 0, \quad k = F, O, P, N, \quad (2.36)$$

with

$$Z_k(-\infty) = Z_{k-\infty}, \quad (2.37)$$

$$Z_k(\infty) = 0, \quad (2.38)$$

for the double-jet problem where $Z_{k-\infty}$ is constant. As a result, all of the Z_k are proportional to each other and to the conserved scalar S which satisfies

$$\frac{d}{dy} \left(\rho D \frac{dS}{dy} \right) - V \frac{dS}{dy} = 0, \quad (2.39)$$

$$S(-\infty) = 1, \quad (2.40)$$

$$S(\infty) = 0, \quad (2.41)$$

where D is a diffusion coefficient.

From (2.36) and (2.39) we can write

$$Z_k = Z_{k-\infty} S(y), \quad k = F, O, P, N. \quad (2.42)$$

Equation (2.39) can be coupled with equations (2.22) and (2.23) to obtain profiles for V , f' and S . To complete the specification of the starting estimate, we must be able to recover the temperature and the major species profiles from the conserved scalar. If we utilize the result in (2.42) along with the fact that in the flame sheet model fuel and oxidizer cannot co-exist, we can derive relations for the temperature and major species on either side of the reaction zone. On the fuel side, we have

$$T = T_{-\infty} S + \left[T_\infty + Y_{O\infty} \frac{Q W_F \nu_F}{c_p W_O \nu_O} \right] (1 - S), \quad (2.43)$$

$$Y_F = Y_{F-\infty} S + Y_{O\infty} \frac{W_F \nu_F}{W_O \nu_O} (S - 1), \quad (2.44)$$

$$Y_O = 0, \quad (2.45)$$

$$Y_P = Y_{O\infty} \frac{W_P \nu_P}{W_O \nu_O} (1 - S), \quad (2.46)$$

and

$$Y_N = Y_{N\infty} (1 - S) + Y_{N-\infty} S. \quad (2.47)$$

On the oxidizer side, we have

$$T = T_\infty (1 - S) + \left[\frac{Q}{c_p} Y_{F-\infty} + T_{-\infty} \right] S, \quad (2.48)$$

$$Y_F = 0, \quad (2.49)$$

$$Y_O = Y_{O\infty} (1 - S) - Y_{F-\infty} \frac{W_O \nu_O}{W_F \nu_F} S, \quad (2.50)$$

$$Y_P = \frac{W_P \nu_P}{W_F \nu_F} Y_{F-\infty} S, \quad (2.51)$$

and

$$Y_N = Y_{N\infty} (1 - S) + Y_{N-\infty} S. \quad (2.52)$$

The diffusion coefficient and the viscosity can be determined by specifying a reference Prandtl number and a transport relation for the viscosity. In particular, we let $(Pr)_{ref} = 0.75$ (air) and

$$\mu = \mu_0 \left(\frac{T}{T_0} \right)^r, \quad (2.53)$$

where $r = 0.7$, $T_0 = 298$ K and $\mu_0 = 1.85 \times 10^{-4}$ gm/cm-sec is again a reference value for air. The scaled heat release parameter $Q/c_p = \Delta T$ can be determined from the heat of combustion of the system and a representative heat capacity.

2.3 Method of Solution

Solution of the governing equations proceeds with an adaptive nonlinear boundary value method. Our goal is to obtain a discrete solution of the governing equations on the mesh \mathcal{M} , where

$$\mathcal{M} = \{-L = y_0 < y_1 < \dots < y_m = L\}. \quad (2.54)$$

With the continuous differential operators replaced by finite difference expressions, we convert the problem of finding an analytic solution of the governing equations to one of finding an approximation to this solution at each point of the mesh \mathcal{M} . We seek the solution U_h^* of the nonlinear system of difference equations

$$F(U_h^*) = 0. \quad (2.55)$$

For an initial solution estimate U^0 that is sufficiently "close" to U_h^* , the system of equations in (2.55) can be solved by Newton's method. We write

$$J(U^k) (U^{k+1} - U^k) = -\lambda_k F(U^k), \quad k = 0, 1, \dots, \quad (2.56)$$

where U^k denotes the k^{th} solution iterate, λ_k the k^{th} damping parameter ($0 < \lambda \leq 1$) and $J(U^k) = \partial F(U^k)/\partial U$ the Jacobian matrix. A system of linear block tridiagonal equations must be solved at each iteration for corrections to the previous solution vector. In the counterflow diffusion flame problem, the cost of forming (we use a numerical Jacobian) and factoring the Jacobian matrix can be a significant part of the cost of the total calculation. In such problems we apply a modified Newton method in which the Jacobian is re-evaluated only periodically [4].

The solution of combustion problems, such as the counterflow diffusion flame, requires that the computational mesh be determined adaptively. Many of the methods that have been used to determine adaptive grids for two-point boundary value problems can be interpreted in terms of equidistributing a positive weight function over a given interval [5,6]. We say that a mesh M is equidistributed on the interval $[-L, L]$ with respect to the non-negative function W and the constant C if

$$\int_{y_j}^{y_{j+1}} W \, dy = C, \quad j = 0, 1, \dots, m-1. \quad (2.57)$$

We determine the mesh by employing a weight function that equidistributes the difference in the components of the discrete solution and its gradient between adjacent mesh points. Upon denoting the vector of $N = K + 3$ dependent solution components by $\tilde{U} = [\tilde{U}_1, \tilde{U}_2, \dots, \tilde{U}_N]^T$, we seek a mesh M such that

$$\int_{y_j}^{y_{j+1}} \left| \frac{d\tilde{U}_i}{dy} \right| dy \leq \delta \left| \max_{-L \leq y \leq L} \tilde{U}_i - \min_{-L \leq y \leq L} \tilde{U}_i \right| \quad \begin{matrix} j = 0, 1, \dots, m-1 \\ i = 1, 2, \dots, N \end{matrix} \quad (2.58)$$

and

$$\int_{y_j}^{y_{j+1}} \left| \frac{d^2\tilde{U}_i}{dy^2} \right| dy \leq \gamma \left| \max_{-L \leq y \leq L} \frac{d\tilde{U}_i}{dy} - \min_{-L \leq y \leq L} \frac{d\tilde{U}_i}{dy} \right| \quad \begin{matrix} j = 1, 2, \dots, m-1 \\ i = 1, 2, \dots, N \end{matrix} \quad (2.59)$$

where δ and γ are small numbers less than one and the maximum and minimum values of \tilde{U}_i and $d\tilde{U}_i/dy$ are obtained from a converged numerical solution on a previously determined mesh.

The coarse to fine grid strategy eliminates many of the convergence difficulties associated with solving the governing equations directly. However, convergence of Newton's method on the initial grid requires a "good" initial estimate for U^0 . We can improve the flame sheet starting estimate by applying a time-dependent starting method on the initial

grid. We remark that, fundamentally, there are two mathematical approaches for solving one-dimensional flame problems - one uses a transient method and the other solves the steady-state boundary value problem directly. Generally speaking, the transient methods are robust but computationally inefficient compared to the boundary value methods, which are efficient but have less desirable convergence properties. Most of the numerical techniques that have been used to solve flame problems have employed a time-dependent method. Variations of this approach have been considered by a variety of researchers (see, e.g., [7-11]). In these methods, the original nonlinear two-point boundary value problem is converted into a nonlinear parabolic mixed initial-boundary value problem. This is accomplished by appending the term $\partial(\bullet)/\partial t$ to the left-hand side of the conservation equations. This results in a semidiscrete set of equations

$$\frac{\partial U}{\partial t} = F(U), \quad (2.60)$$

with appropriate initial conditions. If the time derivative is replaced, for example, by a backward Euler approximation, the governing equations can be written in the form

$$\mathcal{F}(U^{n+1}) = F(U^{n+1}) - \frac{(U^{n+1} - U^n)}{\tau^{n+1}} = 0, \quad (2.61)$$

where for a function $g(t)$ we define $g^n = g(t^n)$ and where the time step $\tau^{n+1} = t^{n+1} - t^n$. At each time level we must solve a system of nonlinear equations that looks very similar to the nonlinear equations in (2.55). Newton's method can again be used to solve this system. The important difference between the system in (2.55) and (2.61) is that the diagonal of the steady-state Jacobian is weighted by the quantity $1/\tau^{n+1}$. This produces a better conditioned system and the solution from the n^{th} time step ordinarily provides an excellent starting guess to the solution at the $(n+1)^{\text{st}}$ time level. The work per time step is similar to that for the modified Newton iteration, but the time-like continuation of the numerical solution produces an iteration strategy that will, in general, be less sensitive to the initial starting estimate than if Newton's method were applied to (2.55) directly. As a result, when we ultimately implement Newton's method on the steady-state equations directly, we obtain a converged numerical solution with only a few additional iterations. This time-dependent starting procedure can also be used on grids other than the initial one.

2.4 Numerical Results

We applied the flame sheet starting estimate to a diluted methane-air flame in the cylindrical double-jet configuration (see Figure 1). As we discussed in Section 2.2, the flame sheet model provides initial solution profiles for the mass flux in the transverse direction, V , the similarity function, f' , the temperature, T , and the major species, i.e.

CH₄, O₂, N₂, CO₂ and CO. Gaussian profiles were used for the minor species. The detailed kinetics mechanism used in the calculations is listed in Table 1.

After getting the flame sheet starting estimate, we solved the full set of governing equations in a two-step procedure. Specifically, we determined a solution to the mass, momentum, and species equations with the energy equation replaced by the flame sheet temperature profile. This procedure is similar to the two-pass solution method used in the solution of adiabatic premixed laminar flames [3]. The fixed flame sheet temperature solution (T_{out}) was then used to obtain a solution to the full fluid dynamic-thermochemistry model (T_{in}). This procedure helped to reduce both convergence difficulties and total CPU time.

In our problem, the separation distance of the two jets was 1.4876 cm. The boundary conditions at the fuel jet were given by

$$V = 2.8 \times 10^{-2}, \quad (2.62)$$

$$f' = 1.216, \quad (2.63)$$

$$Y_{CH_4} = 0.598, \quad Y_{N_2} = 0.402, \quad Y_{k \neq CH_4, N_2} = 0, \quad (2.64)$$

$$T = 294 \text{ K}, \quad (2.65)$$

and at the oxidizer jet by

$$f' = 1.0, \quad (2.66)$$

$$Y_{O_2} = 0.18, \quad Y_{N_2} = 0.82, \quad Y_{k \neq O_2, N_2} = 0, \quad (2.67)$$

$$T = 294 \text{ K}. \quad (2.68)$$

The mass flow rate was in units of gm/cm²-sec and the densities of the fuel and the oxidizer mixtures were used in obtaining the value of the similarity function at the fuel jet. The mass flow rate boundary condition corresponds to a fuel duct velocity of 35 cm/sec. The strain rate used in the calculation was $a = 40 \text{ sec}^{-1}$.

A solution was obtained on a nonuniform grid consisting of 38 grid points. This solution was then used as the starting estimate for the fixed temperature solution. One hundred adaptive time steps were taken to help bring the solution within the domain of convergence of Newton's method on the 38 point grid. After the time steps, Newton's method converged with only one iteration. Once this solution was obtained, the mesh was refined and a solution was calculated on the finer grid. This procedure continued until the adaptive mesh criteria were satisfied. The refined fixed temperature solution was then used as the starting estimate for the complete fluid dynamic-thermochemistry solution. Two additional grid refinements were performed to obtain a final solution on a grid consisting of 65 nonuniform points. On the refined grids Newton's method converged after applying only 10-20 time steps. The mesh spacing was such that 600 equispaced points would have

been needed to obtain comparable accuracy. The total CPU time for the entire procedure was approximately three hours and 40 minutes on a VAX-8600. Approximately 6 CPU seconds were needed for the flame sheet calculation, 130 minutes for the T_{out} calculation and approximately 90 minutes for the T_{in} solution. In contrast, we were unable to obtain a complete T_{in} solution for this problem when the premixed laminar flame starting procedure was employed.

In Figure 2 we compare the flame sheet temperature profile with the calculated finite rate temperature profile. We observe that the agreement is generally quite good. Similar results are illustrated in Figures 3 and 4 for the similarity function and the normal velocity. The flame sheet model is able to predict adequately the location of the stagnation point along with the "double peak" velocity profile.

With the success of the flame sheet starting procedure, we then compared the results of our detailed chemistry model with the experimental measurements of Seshadri [1]. In Figure 5 we illustrate the experimental (circles) and the calculated (solid line) temperature profiles for a problem with similar input conditions as those described in (2.62)-(2.68). We observe that excellent agreement is obtained for the general shape of the profiles as well as for the peak temperatures (maximum experimental = 1691 K and maximum computational = 1680 K). On the rich side of the flame, however, the temperature profiles show some discrepancy. These differences are due most likely to the neglect of C_2 chemistry in the chemical kinetics mechanism. In Figures 6 and 7 we compare the experimental and computational results for the stable major chemical compounds with the physical coordinate as the independent variable. The solid lines represent the numerical calculations and the points represent the experimental measurements. The profiles reveal excellent agreement for the mole fractions of CH_4 , O_2 , N_2 , H_2O and CO_2 . The profiles for CO and H_2 , however, show some discrepancy. The differences between the experiment and the calculations are due again to the neglect of C_2 chemistry in the calculations. Minor species and radicals are illustrated in Figures 8 and 9.

3. FLAME SHEET MODEL OF AN AXISYMMETRIC DIFFUSION FLAME

The modeling of axisymmetric diffusion flames can be reduced to the solution of a set of coupled nonlinear boundary value problems. In these problems we desire solution profiles to as many as several dozen species concentrations in addition to the temperature and the velocity fields. Although axisymmetric flames are important in combustion applications, they have received relatively little attention in theoretical flame studies. Part of this neglect is due to the two-dimensional nature of the problem coupled with the complexities associated with the combined effects of transport phenomena and chemical processes. In the axisymmetric diffusion flame we consider, a fuel jet discharges into a laminar air stream. The flames can be either confined or unconfined. In both cases the tubes through which

the fuel and the oxidizer flow are concentric and have radii R_I and R_O , respectively. In both configurations the two gases make contact at the outlet of the inner tube and a flame resembling a candle results.

In the second part of our Phase I SBIR we investigated the generalization of our one-dimensional flame sheet model to two dimensions. The goal of this work was to provide multi-dimensional starting estimates for the full chemistry-fluid dynamic model that will be studied in Phase II. In an effort that complements the one described in Section 4, we used a stream function-vorticity as opposed to a primitive variable formulation. In this way the pressure (and the continuity equation which is not in the standard elliptic form of the remaining equations) could be eliminated from the model and we could reduce the size of the system to be solved.

We define the vorticity as the amount of counter-clockwise rotation in the fluid. We have

$$\omega = \frac{\partial v_r}{\partial z} - \frac{\partial v_z}{\partial r}. \quad (3.1)$$

The stream function ψ is used to replace the radial and axial components of the velocity vector by a single function. It is defined in such a way that the continuity equation is identically satisfied. We have

$$\rho r v_r = -\frac{\partial \psi}{\partial z}, \quad (3.2a)$$

$$\rho r v_z = \frac{\partial \psi}{\partial r}. \quad (3.2b)$$

Our flame sheet model follows the procedure we discussed in detail in Section 2.2. We again assume a single global irreversible reaction. The diffusion velocity is given by Fick's law and the heat capacities are assumed constant. Our model can be written in the form

$$\frac{\partial}{\partial z} \left(\frac{1}{r\rho} \frac{\partial \psi}{\partial z} \right) + \frac{\partial}{\partial r} \left(\frac{1}{r\rho} \frac{\partial \psi}{\partial r} \right) = -\omega, \quad (3.3)$$

$$\begin{aligned} & r^2 \left[\frac{\partial}{\partial z} \left(\frac{\omega}{r} \frac{\partial \psi}{\partial r} \right) - \frac{\partial}{\partial r} \left(\frac{\omega}{r} \frac{\partial \psi}{\partial z} \right) \right] \\ & - \frac{\partial}{\partial z} \left(r^3 \frac{\partial}{\partial z} \left(\frac{\mu}{r} \omega \right) \right) - \frac{\partial}{\partial r} \left(r^3 \frac{\partial}{\partial r} \left(\frac{\mu}{r} \omega \right) \right) + r^2 g \frac{\partial \rho}{\partial r} + r^2 \nabla \left(\frac{v_r^2 + v_z^2}{2} \right) \cdot \text{iso } \rho = 0, \end{aligned} \quad (3.4)$$

$$\begin{aligned} & - \frac{\partial}{\partial r} \left(Y_F \frac{\partial \psi}{\partial z} \right) + \frac{\partial}{\partial z} \left(Y_F \frac{\partial \psi}{\partial r} \right) - \frac{\partial}{\partial r} \left(r\rho D_F \frac{\partial Y_F}{\partial r} \right) \\ & - \frac{\partial}{\partial z} \left(r\rho D_F \frac{\partial Y_F}{\partial z} \right) - rW_F \nu_F \dot{\omega} = 0, \end{aligned} \quad (3.5)$$

$$\begin{aligned}
& -\frac{\partial}{\partial r} \left(Y_O \frac{\partial \psi}{\partial z} \right) + \frac{\partial}{\partial z} \left(Y_O \frac{\partial \psi}{\partial r} \right) - \frac{\partial}{\partial r} \left(r \rho D_O \frac{\partial Y_O}{\partial r} \right) \\
& - \frac{\partial}{\partial z} \left(r \rho D_O \frac{\partial Y_O}{\partial z} \right) - r W_O \nu_O \dot{w} = 0,
\end{aligned} \tag{3.6}$$

$$\begin{aligned}
& -\frac{\partial}{\partial r} \left(Y_P \frac{\partial \psi}{\partial z} \right) + \frac{\partial}{\partial z} \left(Y_P \frac{\partial \psi}{\partial r} \right) - \frac{\partial}{\partial r} \left(r \rho D_P \frac{\partial Y_P}{\partial r} \right) \\
& - \frac{\partial}{\partial z} \left(r \rho D_P \frac{\partial Y_P}{\partial z} \right) + r W_P \nu_P \dot{w} = 0,
\end{aligned} \tag{3.7}$$

$$\begin{aligned}
& -\frac{\partial}{\partial r} \left(Y_N \frac{\partial \psi}{\partial z} \right) + \frac{\partial}{\partial z} \left(Y_N \frac{\partial \psi}{\partial r} \right) - \frac{\partial}{\partial r} \left(r \rho D_N \frac{\partial Y_N}{\partial r} \right) \\
& - \frac{\partial}{\partial z} \left(r \rho D_N \frac{\partial Y_N}{\partial z} \right) = 0,
\end{aligned} \tag{3.8}$$

and

$$\begin{aligned}
c_p \left[\frac{\partial}{\partial z} \left(T \frac{\partial \psi}{\partial r} \right) - \frac{\partial}{\partial r} \left(T \frac{\partial \psi}{\partial z} \right) \right] - \frac{\partial}{\partial r} \left(r \lambda \frac{\partial T}{\partial r} \right) - \frac{\partial}{\partial z} \left(r \lambda \frac{\partial T}{\partial z} \right) \\
- r \frac{W_F \nu_F h_F + W_O \nu_O h_O - W_P \nu_P h_P}{c_p} \dot{w} = 0,
\end{aligned} \tag{3.9}$$

where \dot{w} is again the rate of progress of the reaction. If we follow the argument in (2.30)-(2.31), we find that each of the Shvab-Zeldovich variables

$$Z_F = Y_F - Y_{F\infty} + \frac{c_p}{Q}(T - T_\infty), \tag{3.10}$$

$$Z_O = Y_O - Y_{O\infty} + \frac{c_p W_O \nu_O}{Q W_F \nu_F}(T - T_\infty), \tag{3.11}$$

$$Z_P = Y_P - Y_{P\infty} - \frac{c_p W_P \nu_P}{Q W_F \nu_F}(T - T_\infty), \tag{3.12}$$

$$Z_N = Y_N - Y_{N\infty}, \tag{3.13}$$

satisfies the differential equation

$$\begin{aligned}
& -\frac{\partial}{\partial r} \left(Z_k \frac{\partial \psi}{\partial z} \right) + \frac{\partial}{\partial z} \left(Z_k \frac{\partial \psi}{\partial r} \right) - \frac{\partial}{\partial r} \left(r \rho D_k \frac{\partial Z_k}{\partial r} \right) \\
& - \frac{\partial}{\partial z} \left(r \rho D_k \frac{\partial Z_k}{\partial z} \right) = 0, \quad k = F, O, P, N.
\end{aligned} \tag{3.14}$$

One can show that all of the Z_k are proportional to each other and to a conserved scalar S that satisfies an equation similar in form to (3.14). Relations for the temperature and the major species follow exactly as in Section 2.2.

We applied the two-dimensional flame sheet model to an unconfined axisymmetric methane-air flame with an inner jet (fuel) radius of 0.254 cm and an outer jet (air) radius

of 2.54 cm. The velocities of the fuel and the oxidizer were 10 cm/sec and the temperature of the two gases was 298 K. Utilizing a generalization of the solution procedure discussed in Section 2.3, we obtained solution profiles for the temperature, major species and the velocities on a 21 x 41 grid. Results of the calculations are illustrated in Figures 10-15.

4. PRIMITIVE VARIABLE METHODS IN TWO DIMENSIONS

In this section we report on experimental modifications to the *TEACH* [12] code for solving the conservation equations of reacting flows in primitive variable form. We examined primarily the CPU-intensive *SIMPLE* (Semi-Implicit Method for Pressure-Linked Equations) algorithm [13] for handling the momentum-pressure coupling, which lies at the heart of *TEACH*, and to which *TEACH* reduces in the laminar, incompressible, single-component case.

TEACH is a multidimensional steady-state compressible flow algorithm based on a low-order control-volume discretization of the primitive variable equations in conservation form over staggered, orthogonal, tensor-product grids. In addition to the continuity and momentum equations, a number (in principle arbitrary) of conserved scalar equations can be accommodated by *TEACH*. These may include an energy equation, multiple specie equations, and the empirical $k - \epsilon$ turbulence model, which supplements the laminar equation system with two extra transport equations of the same generic form while modifying the laminar diffusion terms in the remainder of the equation set.

A large, sparsely coupled set of nonlinear algebraic equations results. The nonlinearities arise directly from the advective terms and the source terms, and indirectly through temperature-, pressure-, and composition-dependent laminar transport properties and thermodynamic coefficients, and the velocity-dependent turbulent transport properties. A multistage variation of the block Gauss-Seidel method is used to solve this nonlinear system. The outermost stage consists of cycling between a Poisson-like pressure correction equation (derived from a truncated substitution of the discrete momentum equations into the discrete continuity equation, which is thus eliminated) and the transport equations for all of the other unknown fields. Within this latter block, the equations are relaxed cyclically, field-by-field. Within the sub-blocks at the level of the individual fields, the updates are computed in a block-line fashion so that a tridiagonal matrix is the largest implicit aggregate in the overall calculation. In practice, under-relaxation of the updates is necessary.

Variations of the *TEACH* code abound. The version with which we make our comparisons is a revision obtained from one of the original authors, as described in [12]. Our principal test problem for this section is the two-dimensional, incompressible, axisymmetric, nonreacting flow in a suddenly expanded laterally heated channel, the hydrodynamics of which are described in [14]. This is the test problem described in [12] and supplied with

the official machine-readable copy of the code, being bound up therewith in a non-modular way.

4.1 Advantages and Disadvantages of *TEACH*

As a base on which to build a detailed-kinetics reacting flow solver, *TEACH* has several advantages, which motivated us to improve its convergence properties:

- The pressure, which is in practice one of the fields of greatest interest, is readily available as one of the unknowns of the problem, whereas the pressure is eliminated in a stream function-vorticity formulation.
- The *TEACH* algorithm generalizes *in principle* to three dimensions, the space in which all real engineering problems lie.
- The widely used $k - \epsilon$ turbulence model is built in.
- The wide user base of *TEACH*-like codes would seem to portend a high interest in the ultimate production version of the reacting flow solver, and relative ease of transferability to the interested community.

There are also some major disadvantages:

- *TEACH* has many parameters (such as underrelaxation factors and iteration limits) that are difficult to tune efficiently, and poor diagnostics by which to determine "convergence".
- Convergence is slow, being asymptotically linear at best, even for "easy" problems (those with few species, the conservation equations for which are dominated by the linearly implicit part of the source term, or by advection or diffusion).
- Convergence is unreliable in "difficult" problems (those with many species, the conservation equations for which are dominated by parts of the source term not treated implicitly).
- There are no convenient data structures built into *TEACH* to assist the user in organizing the large volume of physical data (kinetic, transport, and thermodynamic) which must be supplied in realistic detailed modeling.

Because of these disadvantages, much of our work in preparation for detailed modeling in two dimensions has been carried out in a context other than the *TEACH* code, as described above. However, since there is ultimate interest in a primitive variables staggered grid code, we addressed the first two disadvantages to *TEACH*, as described in the balance of this section.

4.2 Modifications of the *TEACH* Code – Outer Loop

The weakness at the heart of the *TEACH* algorithm is in the solution of the discrete nonlinear system. Linearization by decoupling at the field-by-field level provides only a linear convergence rate for the outer iteration. Furthermore, the degree of underrelaxation found necessary in practice causes this rate to be slow. The most natural modification was to implement a Newton-like algorithm for the outer loop.

Newton methods are desirable for their asymptotically superlinear convergence rates. They have the big disadvantage compared to simpler iterative schemes, such as that of plain *TEACH*, that they require formation and factorization of a Jacobian matrix. We examined a recently developed Jacobian-free Newton-like method known as the nonlinear generalized minimum residual method (NLGMR). A precursor of this method may be found in [15]. More recently, very impressive results have been obtained with NLGMR in the computational fluid dynamics context [16]. The incorporation of NLGMR into *TEACH* can be accomplished in a very modular fashion, leaving most of the original code fully intact. To be specific, the NLGMR algorithm is inserted between the control driver of *TEACH* and the calls to the field-by-field solvers.

Let the action of one outer iteration of the *TEACH* algorithm be denoted by

$$u_{n+1} \leftarrow M(u_n), \quad (4.1)$$

where the vector $u \in \mathbf{R}^N$ represents the discrete unknowns of all of the fields (u , v , p , etc.) and M is the nonlinear mapping that produces the $(n+1)$ st iterate from the n th. In the *TEACH* code, M represents one pass through the routines *CALCU*, *CALCV*, *CALCP*, etc., in their proper cyclic order. The converged solution u_* is the root of the system of nonlinear equations

$$F(u) \equiv u - M(u) = 0. \quad (4.2)$$

With the specification of the initial iterate, u_0 , the *TEACH* algorithm has the form of a nonlinear block Gauss-Seidel method for $F(u) = 0$.

NLGMR provides a means of accelerating the *TEACH* iterations as follows (see [16]). Throughout we denote by $J(u)$ the Jacobian matrix of F evaluated at u . When there is no ambiguity, $J(u)$ will be denoted simply by J . Suppose that u_n is the current approximation and that we wish to find a new approximation of the form $u_{n+1} = u_n + \delta$.

We write δ in the form

$$\delta = \sum_{i=1}^m \alpha_i v_i, \quad (4.3)$$

where the $v_i \in \mathbf{R}^N$ are m orthonormal vectors, the specification of which will be given shortly, and the α_i are unknowns to be determined at each step. Ideally, we would carry out the minimization of

$$\|F(u_n + \delta)\|$$

over all vectors δ to get the new iterate $u_{n+1} = u_n + \delta$. Instead of solving this numerically difficult problem directly, we can linearize F about u_n by writing $F(u_n + \delta) \approx F(u_n) + J(u_n)\delta$ and seek to minimize

$$\|F(u_n) + J\delta\|$$

over all vectors δ of the form (4.3). If we choose the v_i to form an orthonormal basis for the Krylov subspace based on the Jacobian, namely $\text{span} \{v_1, Jv_1, J^2v_1, \dots, J^{m-1}v_1\}$, where $v_1 = F(u_n)/\|F(u_n)\|$, then this minimization can be carried out by performing m steps of the GMRES algorithm [17] applied to the linear system

$$J\delta = -F(u_n), \quad (4.4)$$

with the initial iterate $\delta_0 = 0$. Note that an exact solution of (4.4) would yield the Newton direction $J^{-1}F(u_n)$. We leave most of the details of GMRES to the references; however, we note several of its important properties. If $m = N$ the method, though iterative in nature, delivers the exact solution of (4.4) and is thus a strict Newton's method in this limit. If the Jacobian is suitably conditioned, however, GMRES usually converges for $m \ll N$, resulting in considerable savings over the linear algebra expense of a direct elimination Newton algorithm. For sufficiently large m , its storage requirements are roughly half those of theoretically equivalent Krylov subspace methods, such as GCR and ORTHODIR [18], and its CPU time requirements are about one-third less. Finally, the method cannot break down before finding the solution (in the absence of roundoff error) for arbitrary nonsymmetric indefinite matrices J .

Perhaps the most important aspect of the NLGMR algorithm described above is that the Jacobian matrix J is never needed explicitly. The only operations using the Jacobian matrix J that are actually required in the implementation are matrix-vector multiplications of the form $w = Jv$, which can be approximated by finite differences of F , viz.,

$$J(u)v \approx \frac{F(u + hv) - F(u)}{h}, \quad (4.5)$$

where u is the point where the Jacobian is being evaluated and h is some carefully chosen small scalar. In comparison to the typical cost of evaluating $F(u)$, which requires one application of the mapping $M(u)$, there is very little overhead in the NLGMR algorithm, and its overall cost can be estimated in units of F -evaluations, which are the same as plain *TEACH* outer iterations.

An unfortunate and well-known flaw of Newton's method is that the domain of convergence is typically quite small. In practice, NLGMR invariably diverges when applied to *TEACH* in the form (4.2). In contrast, the *TEACH* iterations can often be stabilized by sufficient underrelaxation if one can afford the price of the concomitant slow convergence. A reliable cure for NLGMR, following [16], is to introduce damping into $F(u)$, that is, to replace (4.2) at each step with

$$F(u) = u - (1 - \lambda)M(u) - \lambda M(u_n), \quad (4.6)$$

where λ is a damping parameter to be selected adaptively, $0 \leq \lambda \leq 1$. Selecting $\lambda = 0$ corresponds to Newton's method. Selecting $\lambda = 1$ corresponds to ordinary, unaccelerated *TEACH*. If λ is held at 1 until the *TEACH* iteration residuals begin to exhibit monotonic convergence, a subsequent power-law decrease in λ (as a function of iteration count) generally brings about asymptotically superlinear convergence. To be specific, let ρ_n denote some scaled Euclidean norm of the components of $F(u_n)$, and let \bar{n} denote the iteration level at which the monotonic residual decrease criterion is first satisfied. (In practice, we have been requiring an absolute decrease for five successive iterations, because of the notorious oscillatory convergence profiles of plain *TEACH*.) Let ρ_n denote the residual at iteration \bar{n} . Thereafter, we take

$$\lambda_n = (\rho_n/\rho_{\bar{n}})^2 \cdot b^{n-\bar{n}}, \quad (4.7)$$

where b is a tuning parameter governing the rate of decrease of λ , supplied by the user. Typically, we choose b in the range 0.7-0.8, which promotes a fairly rapid transition from plain *TEACH* to Newton's method once the tail of the *TEACH* iterations has been reached. The pre-exponential factor is designed to hasten adaptively the transition to Newton's method according to the rate of decrease of the residual.

In Figures 16-19 we show the type of savings that are achievable by wrapping NLGMR around *TEACH* in this manner. The four figures are in two pairs—one pair (Figures 16 and 17) for a 16×16 grid (14 interior cells), and one (Figures 18 and 19) for a 30×30 grid (28 interior cells), or twice the resolution of the first. Each pair contrasts the convergence history of plain and NLGMR-accelerated *TEACH* for a laminar version of the standard test problem by plotting the mass and momentum equation residuals (solid curves – individually nondimensionalized as in the original *TEACH*) and the composite NLGMR residual (dashed curve) against the number of evaluations of $M(u)$. In the 16×16 case, the monotonically decreasing tail is reached at 33 function evaluations. Thereafter, plain *TEACH* requires 105 outer iterations (costing one function evaluation each) to reach the convergence criterion of a relative residual reduction of 10^{-8} , whereas NLGMR-accelerated *TEACH* requires only 7 additional outer iterations (costing a total of 67 function evaluations). Each plateau in the NLGMR plots represents one outer iteration during which several GMRES steps may be required. In the 30×30 case, the monotonically decreasing tail is reached at 60 function evaluations. Thereafter, plain *TEACH* requires 153 outer iterations (costing one function evaluation each) to reach the convergence criterion of a relative residual reduction of 10^{-8} , whereas NLGMR-accelerated *TEACH* requires only 9 additional outer iterations (costing a total of 108 function evaluations). In terms of the overall number of function evaluations from start to finish NLGMR allows reductions of 28% and 21%, respectively. In no case was a Krylov subspace of dimension higher than 20 required.

This is not a dramatic improvement; on the other hand, it has been achieved with virtually no extra programming cost to the user, and some possible side benefits of NLGMR

are as yet unexplored in connection with *TEACH*. For instance, in [16] the authors found it possible to play loosely with the tuning parameters of an algorithm being accelerated by NLGMR, since NLGMR had a stabilizing effect on the overall iteration. Hence, some of the uncertainty that accompanies the selection of *TEACH* parameters for problems of a previously unexplored scale may be relieved.

We conclude that the damped NLGMR technique provides a useful enhancement to plain *TEACH* in its present state of development. Further efforts to successfully marry these two algorithms (exogenous to our AFOSR-sponsored reacting flow work) are currently underway by SCA consultants.

4.3 Modifications of the *TEACH* Code – Inner Loop

In view of the preservation of the *TEACH* inner loops (as embodied in the routines CALCU, CALCV, CALCP, etc.) in the NLGMR-enhanced version described above, it was natural to look for further savings in solving the system of linear equations for the updates to each field. Labeling each of u , v , p , etc. by the generic field ϕ , in turn, these linear systems at each inner iteration of the *TEACH* algorithm nominally have the form

$$a_P \phi_P = \sum_k a_k \phi_k + s_P, \quad (4.8)$$

at each point P of the grid on which ϕ is defined, where the sum extends over the four neighbors of P . The coefficients arise from the control-volume discretization of the advection, diffusion and source operators using the “best available” velocity and property data. Because the advective fluxes are of hybrid upwinded form, all of the a_k are nonnegative. Because of the conservation form of the discretization,

$$a_P = \sum_k a_k. \quad (4.9)$$

In fact, the field updates are generally underrelaxed due to the nonlinearities hidden in the coefficients of (4.8). Thus, if ϕ^* represents the vector of unknowns calculated from (4.8), where the coefficients are based on the n th iterate ϕ^n , then

$$\phi^{n+1} = \mu \phi^* + (1 - \mu) \phi^n, \quad (4.10)$$

where μ , $0 \leq \mu \leq 1$ is an underrelaxation parameter, which may be set differently for each field. (Typical values of μ are 0.5 for the velocities and 1.0 for the pressure, with the other conserved scalars somewhere in between.) If the degree of underrelaxation desired in (4.10) is known *a priori*, it may be built into the coefficients of a modified version of (4.8) with advantage. In fact, the *TEACH* routine block-line solver LISOLV actually solves

$$a_P^* \phi_P = \sum_k a_k \phi_k + s_P^*, \quad (4.11)$$

directly for ϕ^{n+1} , where $a_p^* \equiv a_p/\mu$ and $s_p^* \equiv s_p + (1 - \mu)a_p^*\phi^n$. This is algebraically equivalent to the two-step process (4.8)-(4.9), but is numerically advantageous, because it increases the diagonal dominance of the linear system, thus enhancing the convergence rate of the block-line iterative solver.

For systems of appreciable size, GMRES with some form of approximate factorization preconditioner is among the best methods for problems of the form (4.8). Hence, the block-line solver LISOLV was replaced with several differently preconditioned GMRES methods from *PCGPAK* [19]. The results were disappointing, with GMRES typically requiring 2.0 to 2.5 times as much CPU time as the block-line solver for the same degree of residual reduction on a 16×16 grid with the full turbulent test problem mechanism in place. The conclusion of this brief exercise is that for the amount of underrelaxation already built into *TEACH* to stabilize the outer nonlinear iterations, the most elementary linear solvers perform most efficiently for the inner iterations. Therefore, we recommend retaining LISOLV in conjunction with NLGMR-*TEACH*.

5. CONCLUSIONS

We have demonstrated the feasibility of combining detailed transport phenomena with complex chemistry in the solution of chemically reacting flows. We have illustrated the effectiveness of flame sheet initialization procedures and a time integration-Newton nonlinear equation solver in the modeling of a counterflow diffusion flame. We have also made improvements to the outer iteration of the primitive variable *TEACH* code. Our goals in a Phase II SBIR will be to generalize these ideas to enable the efficient solution of axisymmetric (laminar and turbulent) diffusion flames. Based upon our Phase I results, we are confident that our approach will be successful.

List of Tables

1. Reaction Mechanism for the Methane-air Flame.

List of Figures

1. Schematic illustration of a counterflow flame.
2. Comparison between the calculated flame sheet temperature profile (dotted line) and the calculated finite rate chemistry temperature profile (solid line) in the double-jet problem.
3. Comparison between the calculated flame sheet similarity function profile (dotted line) and the calculated finite rate chemistry similarity function profile (solid line) in the double-jet problem.
4. Comparison between the calculated flame sheet normal velocity profile (dotted line) and the calculated finite rate chemistry normal velocity profile (solid line) in the double-jet problem.
5. Comparison between measured (\circ) and calculated values (solid line) of the temperature profile.
6. Comparison between measured CH_4 (\square), O_2 (\circ) and N_2 (\diamond) profiles and corresponding calculated values (solid lines).
7. Comparison between measured H_2O (\square), CO_2 (\circ), H_2 (+) and CO (\diamond) profiles and corresponding calculated values (solid lines).
8. Calculated profiles of the major reactive species and radicals in the flame.
9. Calculated profiles of the trace reactive species and radicals in the flame.
10. Flame sheet temperature distribution in a two-dimensional axisymmetric diffusion flame.
11. Flame sheet CH_4 (mass fraction) distribution in a two-dimensional axisymmetric diffusion flame.
12. Flame sheet O_2 (mass fraction) distribution in a two-dimensional axisymmetric diffusion flame.
13. Flame sheet H_2O (mass fraction) distribution in a two-dimensional axisymmetric diffusion flame.
14. Flame sheet CO_2 (mass fraction) distribution in a two-dimensional axisymmetric diffusion flame.

15. Velocity (vector) distribution in a two-dimensional axisymmetric diffusion flame.
16. Residuals plotted against number of function evaluations (approximately proportional to CPU time) for Plain *TEACH* for the test problem on a 16×16 grid.
17. Residuals plotted against number of function evaluations for *NLGMR TEACH* for the test problem on a 16×16 grid.
18. Residuals plotted against number of function evaluations for Plain *TEACH* for the test problem on a 30×30 grid.
19. Residuals plotted against number of function evaluations for *NLGMR TEACH* for the test problem on a 30×30 grid.

TABLE I.

Reaction Mechanism Rate Coefficients In The Form $k_f = AT^\beta \exp(-E_0/RT)$.

Units are moles, cubic centimeters, seconds, Kelvins and calories/mole.

	REACTION	A	β	E
1.	$CH_4 + M = CH_3 + H + M$	1.00E+17	0.000	86000.
2.	$CH_4 + O_2 = CH_3 + HO_2$	7.90E+13	0.000	56000.
3.	$CH_4 + H = CH_3 + H_2$	2.20E+04	3.000	8750.
4.	$CH_4 + O = CH_3 + OH$	1.60E+06	2.360	7400.
5.	$CH_4 + OH = CH_3 + H_2O$	1.60E+06	2.100	2460.
6.	$CH_2O + OH = HCO + H_2O$	7.53E+12	0.000	167.
7.	$CH_2O + H = HCO + H_2$	3.31E+14	0.000	10500.
8.	$CH_2O + M = HCO + H + M$	3.31E+16	0.000	81000.
9.	$CH_2O + O = HCO + OH$	1.81E+13	0.000	3082.
10.	$HCO + OH = CO + H_2O$	5.00E+12	0.000	0.
11.	$HCO + M = H + CO + M$	1.60E+14	0.000	14700.
12.	$HCO + H = CO + H_2$	4.00E+13	0.000	0.
13.	$HCO + O = OH + CO$	1.00E+13	0.000	0.
14.	$HCO + O_2 = HO_2 + CO$	3.00E+12	0.000	0.
15.	$CO + O + M = CO_2 + M$	3.20E+13	0.000	-4200.
16.	$CO + OH = CO_2 + H$	1.51E+07	1.300	-758.
17.	$CO + O_2 = CO_2 + O$	1.60E+13	0.000	41000.
18.	$CH_3 + O_2 = CH_3O + O$	7.00E+12	0.000	25652.
19.	$CH_3O + M = CH_2O + H + M$	2.40E+13	0.000	28812.
20.	$CH_3O + H = CH_2O + H_2$	2.00E+13	0.000	0.
21.	$CH_3O + OH = CH_2O + H_2O$	1.00E+13	0.000	0.
22.	$CH_3O + O = CH_2O + OH$	1.00E+13	0.000	0.
23.	$CH_3O + O_2 = CH_2O + HO_2$	6.30E+10	0.000	2600.
24.	$CH_3 + O_2 = CH_2O + OH$	5.20E+13	0.000	34574.
25.	$CH_3 + O = CH_2O + H$	6.80E+13	0.000	0.
26.	$CH_3 + OH = CH_2O + H_2$	7.50E+12	0.000	0.
27.	$HO_2 + CO = CO_2 + OH$	5.80E+13	0.000	22934.
28.	$H_2 + O_2 = 2OH$	1.70E+13	0.000	47780.
29.	$OH + H_2 = H_2O + H$	1.17E+09	1.300	3626.
30.	$H + O_2 = OH + O$	5.13E+16	-0.816	16507.

TABLE I. (continued)

Reaction Mechanism Rate Coefficients In The Form $k_f = AT^\beta \exp(-E_0/RT)$.

Units are moles, cubic centimeters, seconds, Kelvins and calories/mole.

	REACTION	A	β	E
31.	$O + H_2 = OH + H$	1.80E+10	1.000	8826.
32.	$H + O_2 + M = HO_2 + M^a$	2.10E+18	-1.000	0.
33.	$H + O_2 + O_2 = HO_2 + O_2$	6.70E+19	-1.420	0.
34.	$H + O_2 + N_2 = HO_2 + N_2$	6.70E+19	-1.420	0.
35.	$OH + HO_2 = H_2O + O_2$	5.00E+13	0.000	1000.
36.	$H + HO_2 = 2OH$	2.50E+14	0.000	1900.
37.	$O + HO_2 = O_2 + OH$	4.80E+13	0.000	1000.
38.	$2OH = O + H_2O$	6.00E+08	1.300	0.
39.	$H_2 + M = H + H + M^b$	2.23E+12	0.500	92600.
40.	$O_2 + M = O + O + M$	1.85E+11	0.500	95560.
41.	$H + OH + M = H_2O + M^c$	7.50E+23	-2.600	0.
42.	$H + HO_2 = H_2 + O_2$	2.50E+13	0.000	700.
43.	$HO_2 + HO_2 = H_2O_2 + O_2$	2.00E+12	0.000	0.
44.	$H_2O_2 + M = OH + OH + M$	1.30E+17	0.000	45500.
45.	$H_2O_2 + H = HO_2 + H_2$	1.60E+12	0.000	3800.
46.	$H_2O_2 + OH = H_2O + HO_2$	1.00E+13	0.000	1800.

^a Third body efficiencies: $k_5(H_2O) = 21k_5(Ar)$, $k_5(H_2) = 3.3k_5(Ar)$, $k_5(N_2) = k_5(O_2) = 0$.^b Third body efficiencies: $k_{12}(H_2O) = 6k_{12}(Ar)$, $k_{12}(H) = 2k_{12}(Ar)$, $k_{12}(H_2) = 3k_{12}(Ar)$.^c Third body efficiency: $k_{14}(H_2O) = 20k_{14}(Ar)$.

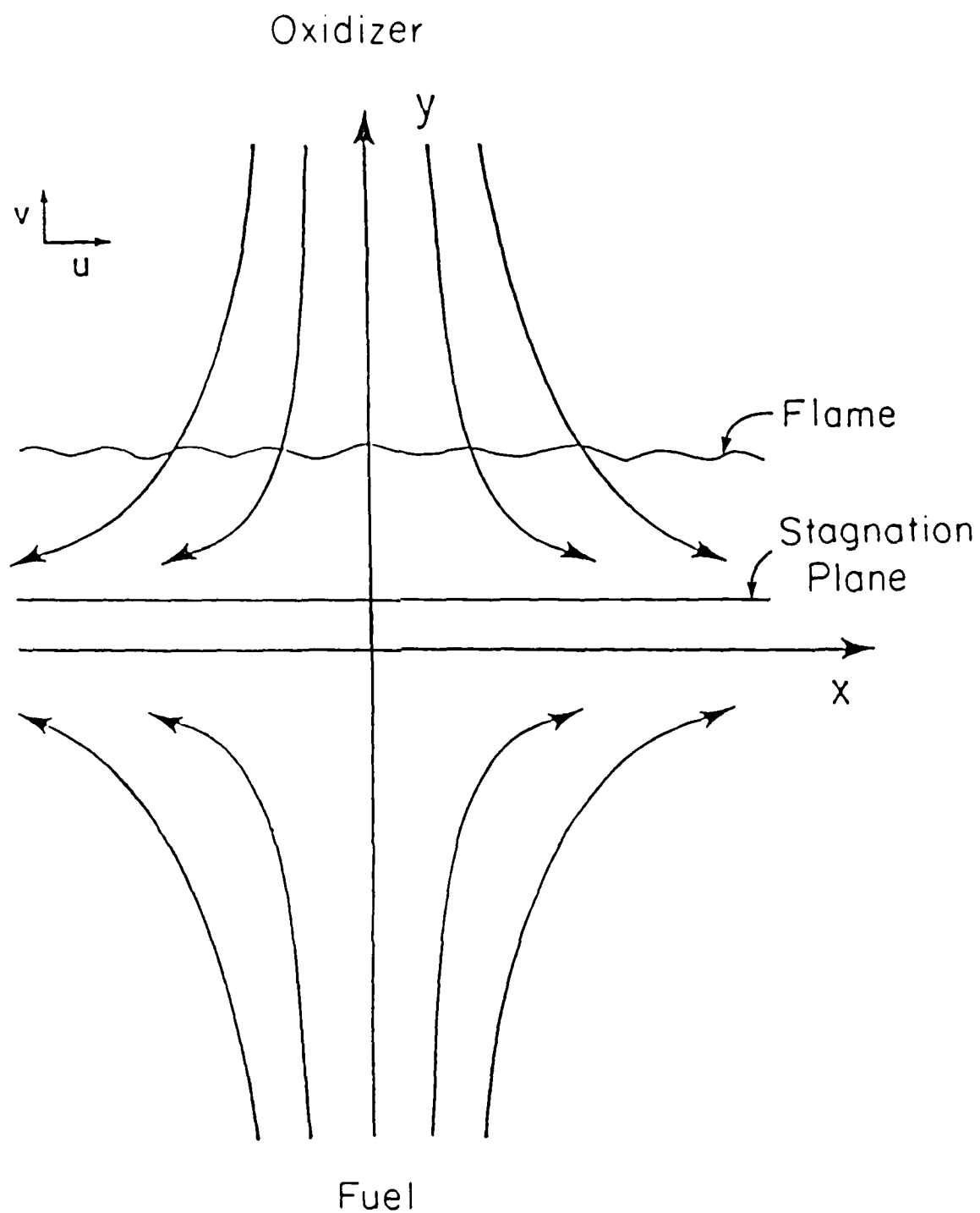


Figure 1: Schematic illustration of a counterflow flame.

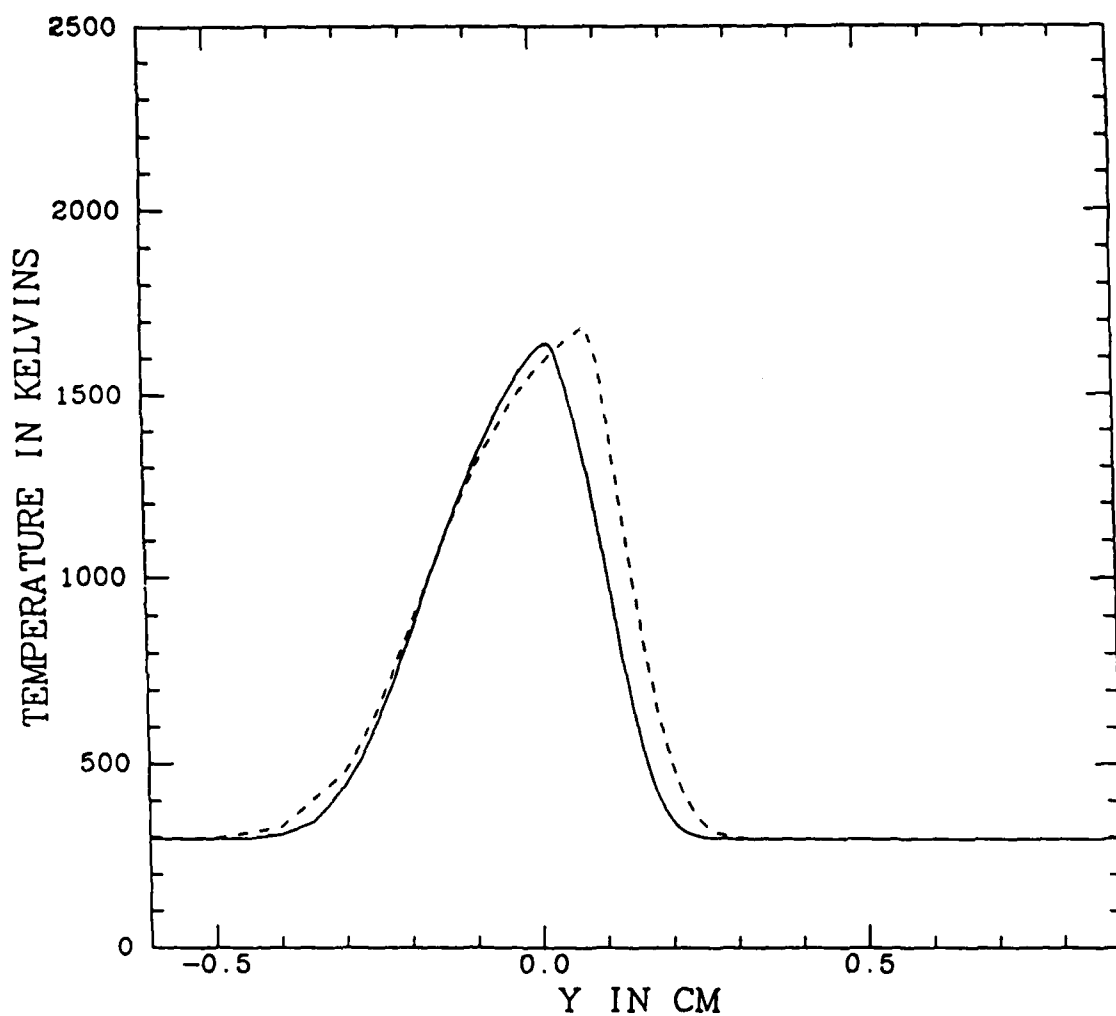


Figure 2: Comparison between the calculated flame sheet temperature profile (dotted line) and the calculated finite rate chemistry temperature profile (solid line) in the double-jet problem.

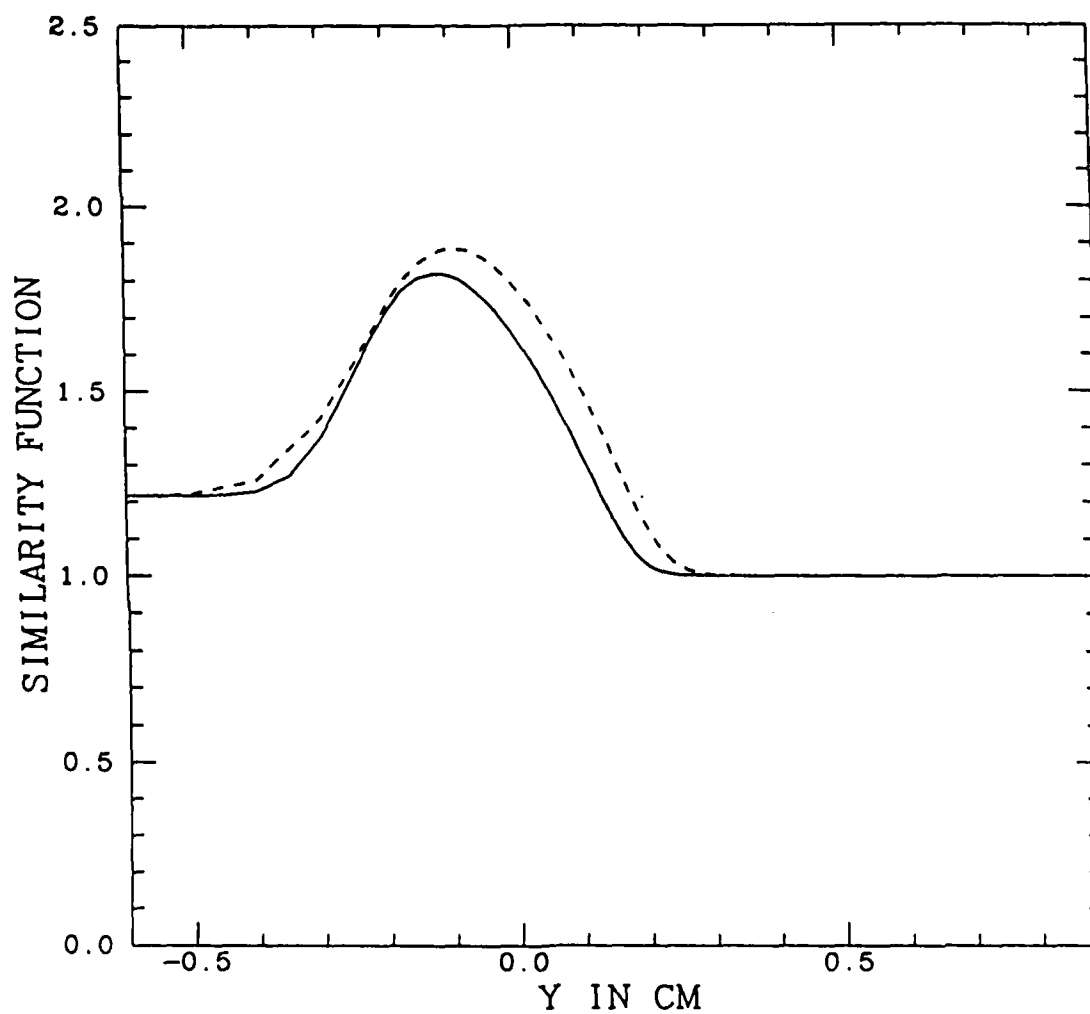


Figure 3: Comparison between the calculated flame sheet similarity function profile (dotted line) and the calculated finite rate chemistry similarity function profile (solid line) in the double-jet problem.

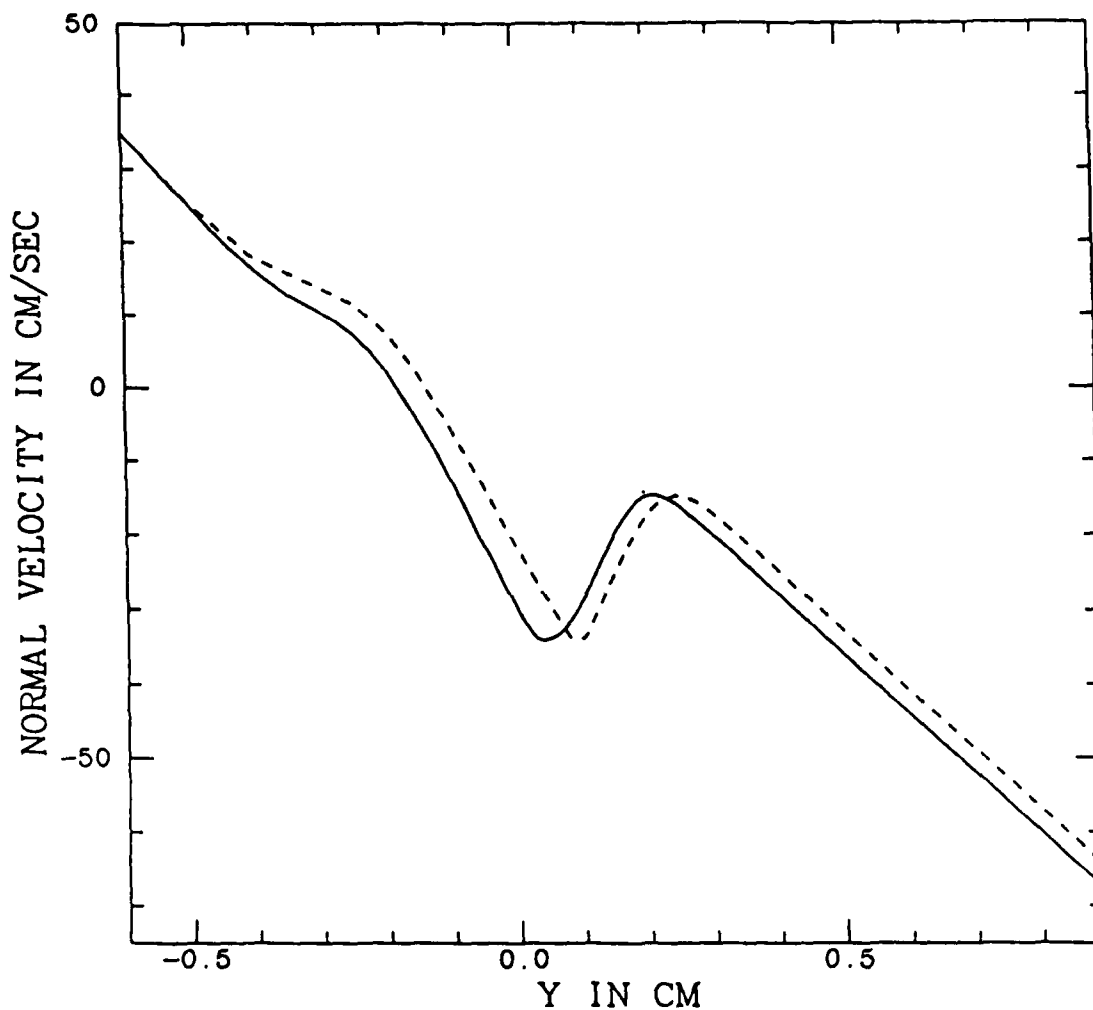


Figure 4: Comparison between the calculated flame sheet normal velocity profile (dotted line) and the calculated finite rate chemistry normal velocity profile (solid line) in the double-jet problem.

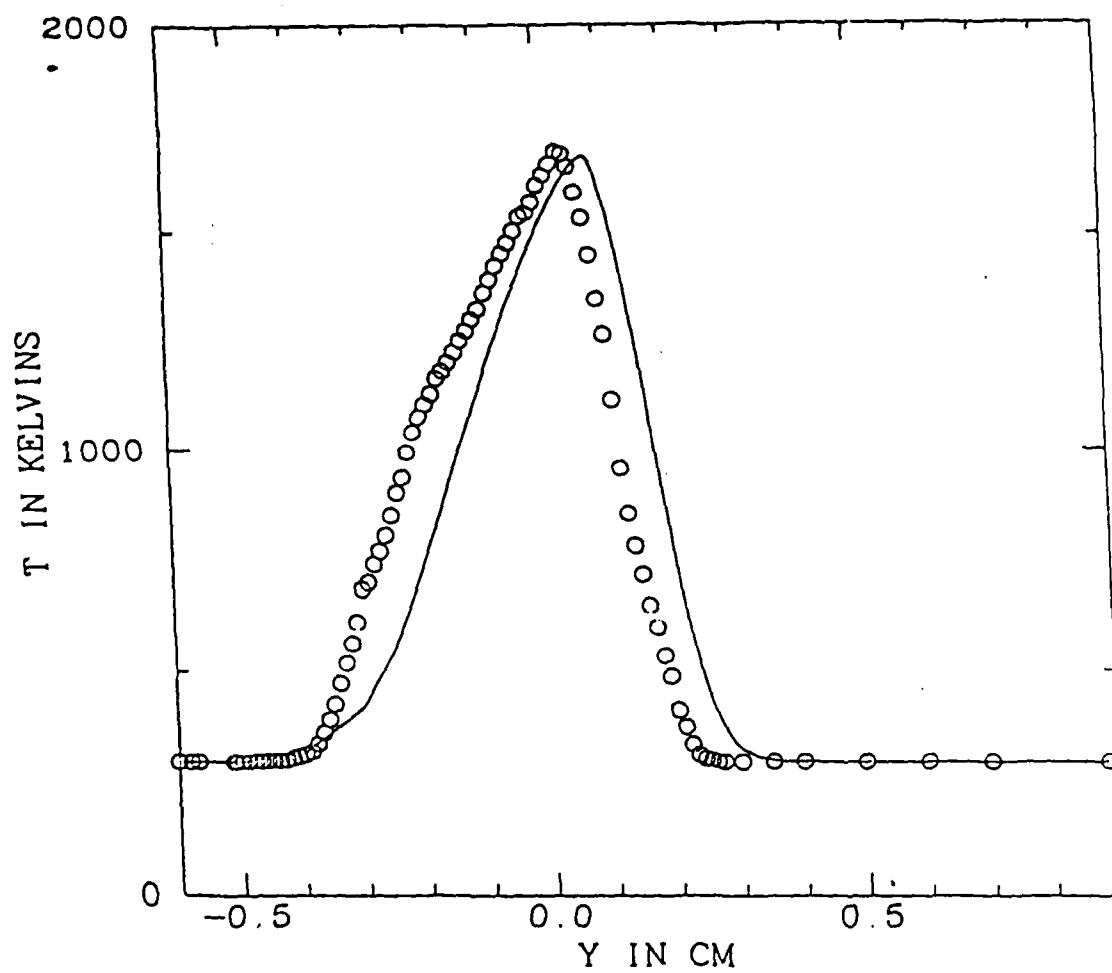


Figure 5: Comparison between measured (o) and calculated values (solid line) of the temperature profile.

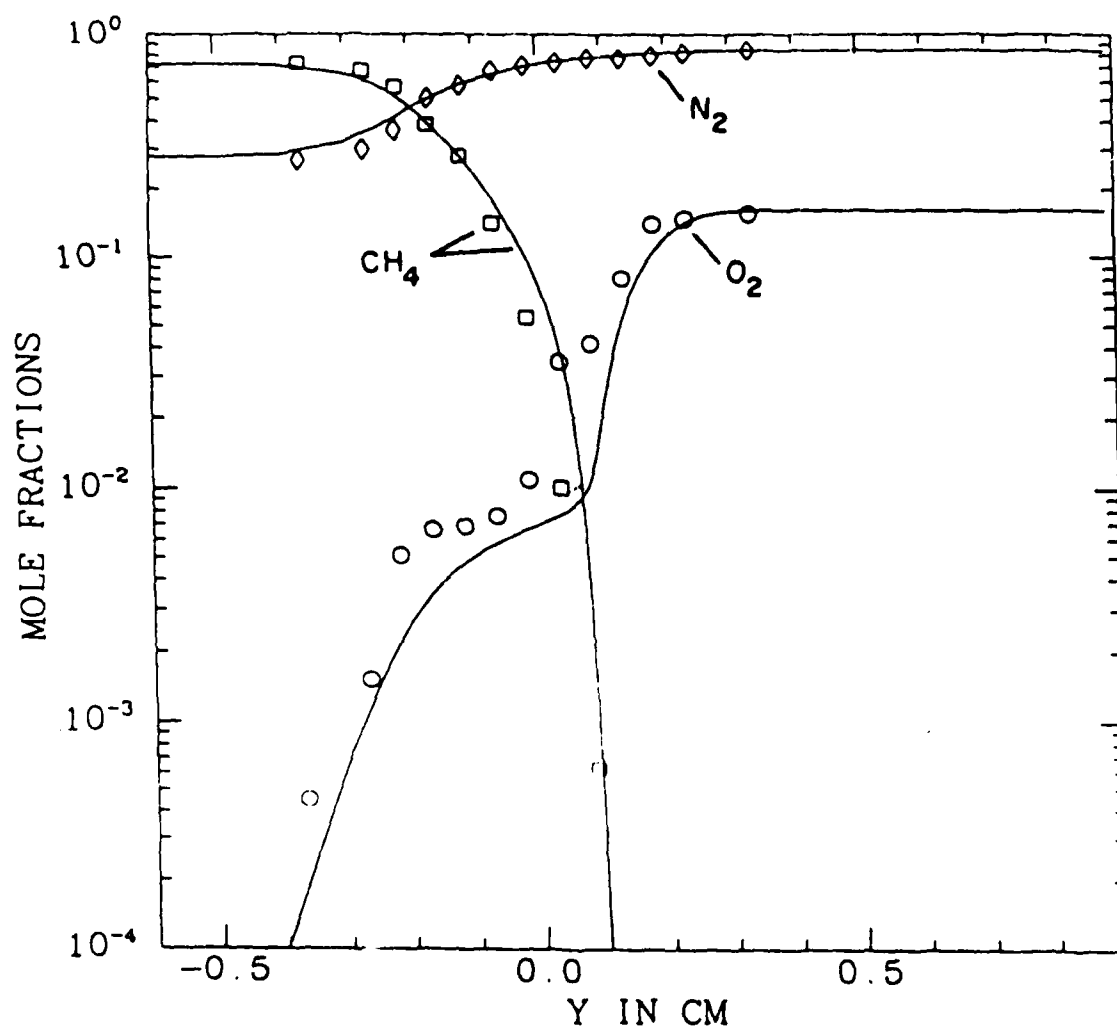


Figure 6: Comparison between measured CH_4 (\square), O_2 (\circ) and N_2 (\diamond) profiles and corresponding calculated values (solid lines).

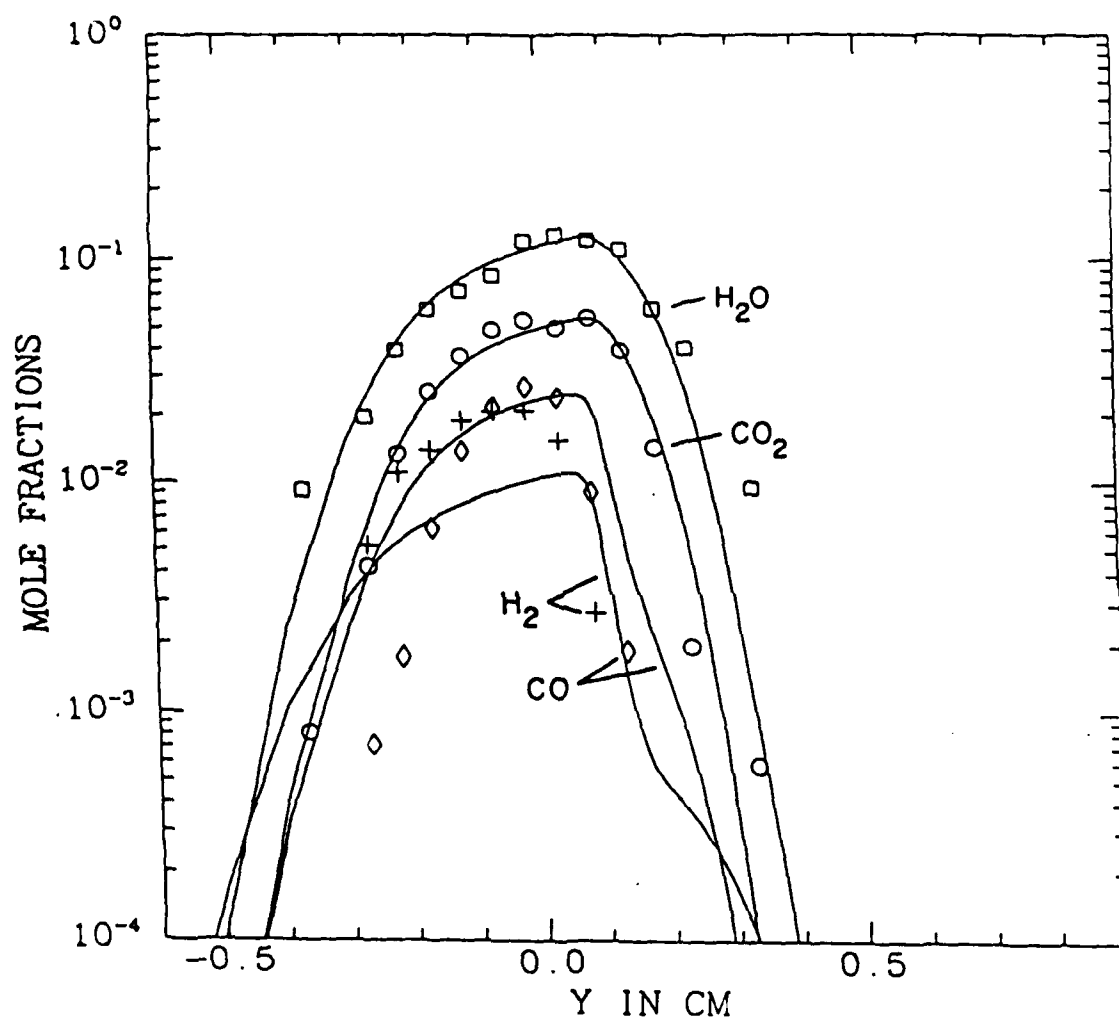


Figure 7: Comparison between measured H_2O (\square), CO_2 (\circ), H_2 (+) and CO (\diamond) profiles and corresponding calculated values (solid lines).

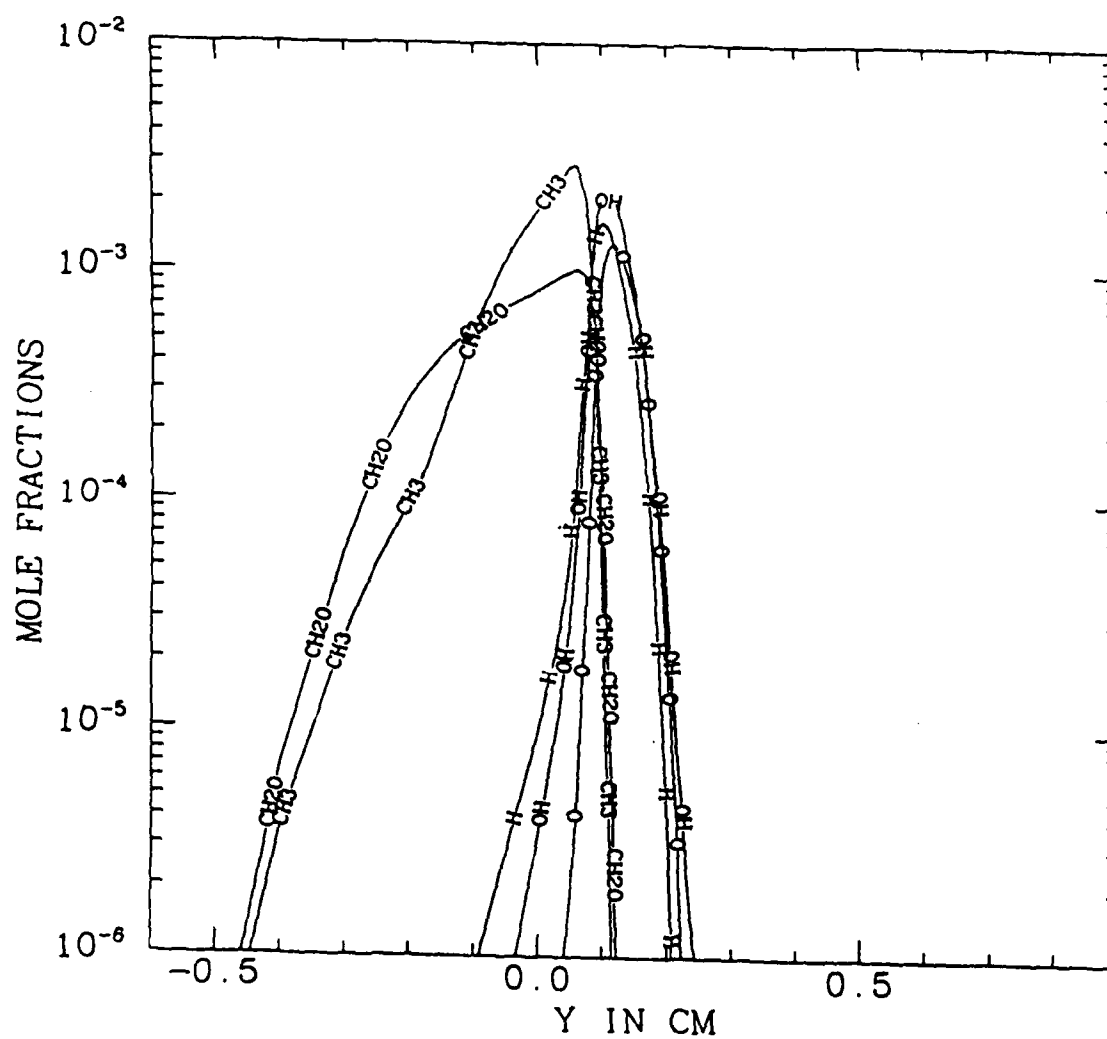
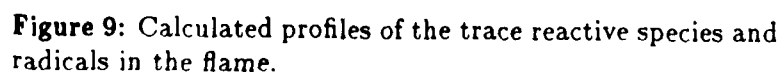


Figure 8: Calculated profiles of the major reactive species and radicals in the flame.



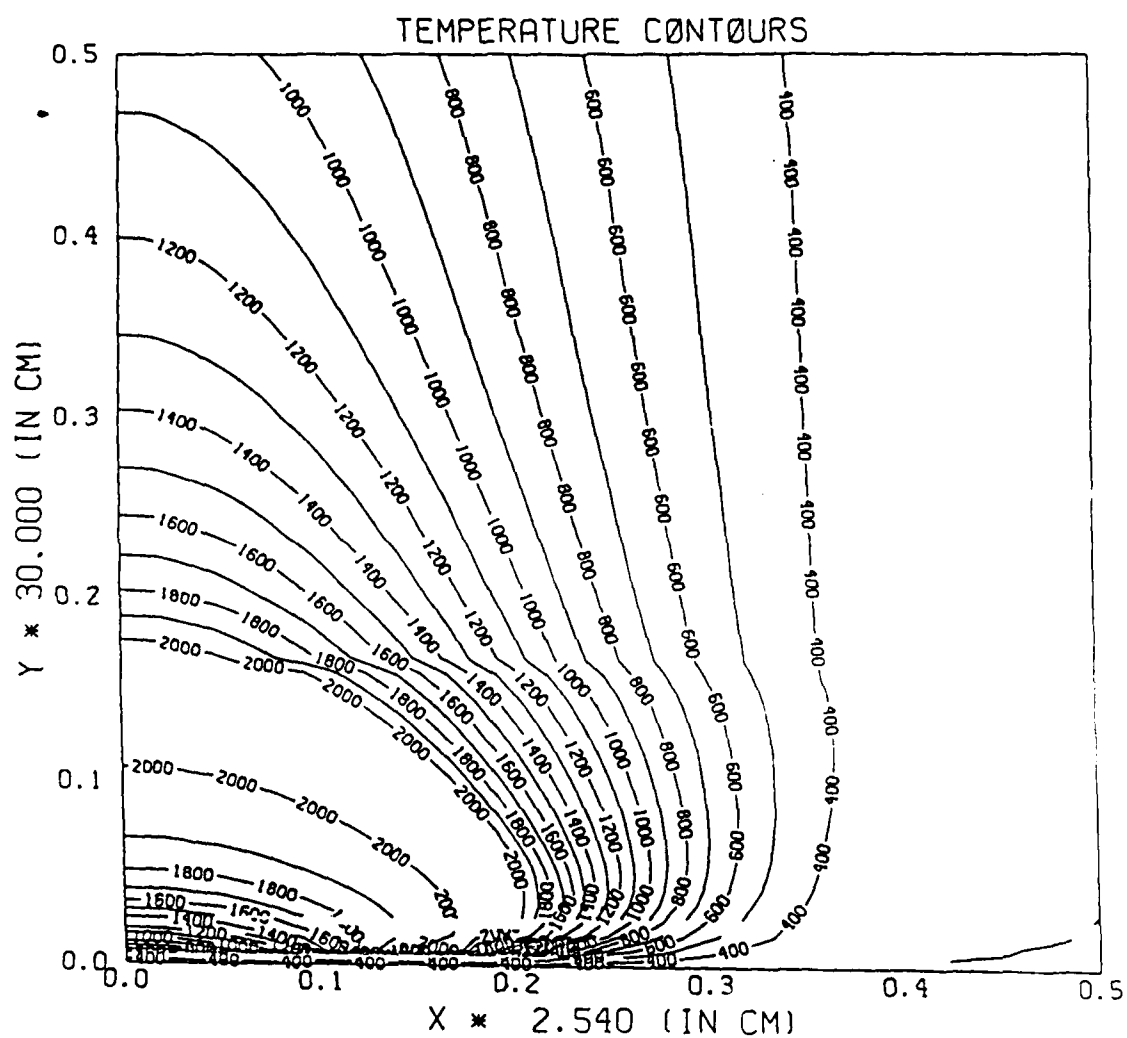


Figure 10: Flame sheet temperature distribution in a two-dimensional axisymmetric diffusion flame.

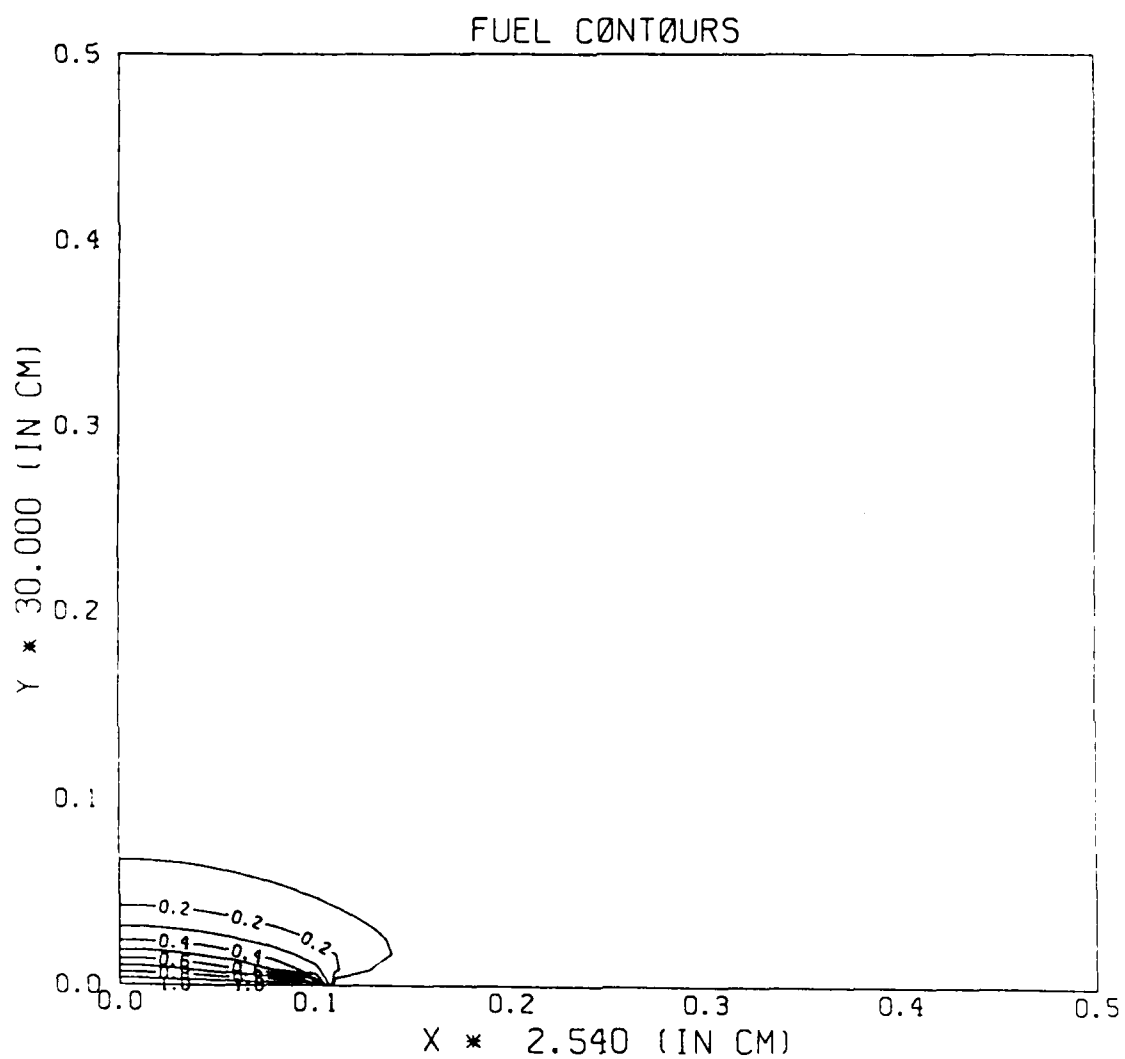


Figure 11: Flame sheet CH_4 (mass fraction) distribution in a two-dimensional axisymmetric diffusion flame.

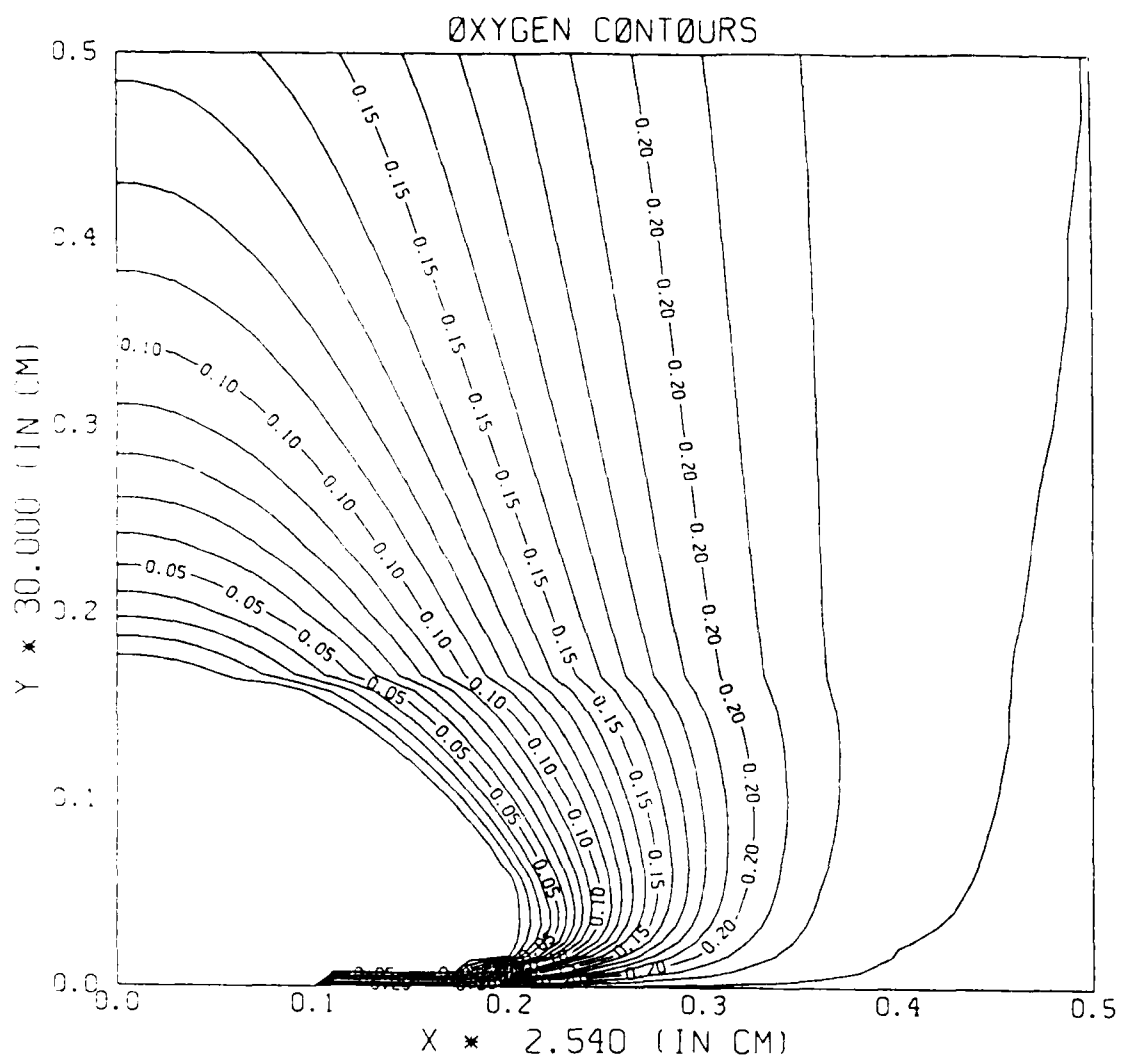


Figure 12: Flame sheet O_2 (mass fraction) distribution in a two-dimensional axisymmetric diffusion flame.

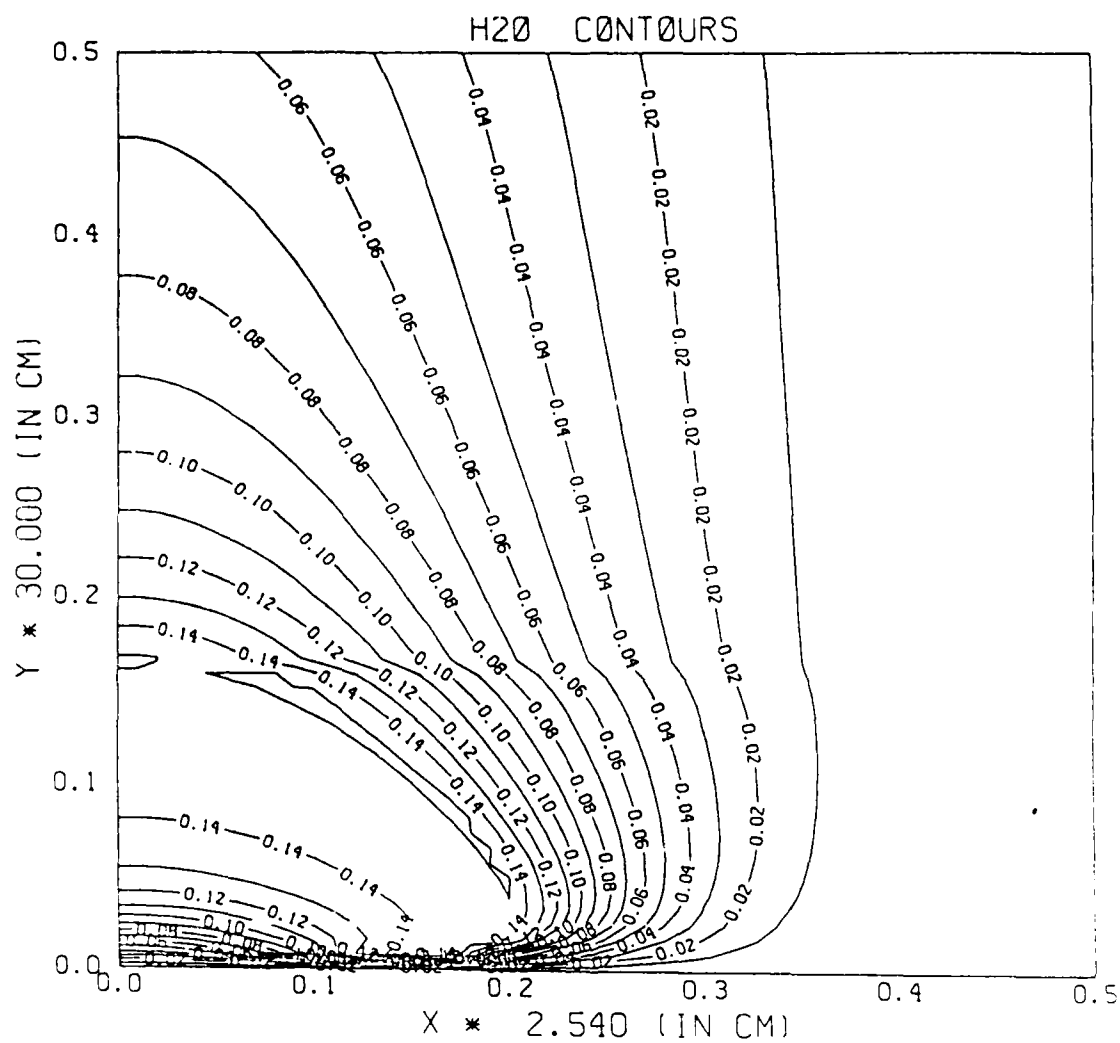


Figure 13: Flame sheet H_2O (mass fraction) distribution in a two-dimensional axisymmetric diffusion flame.

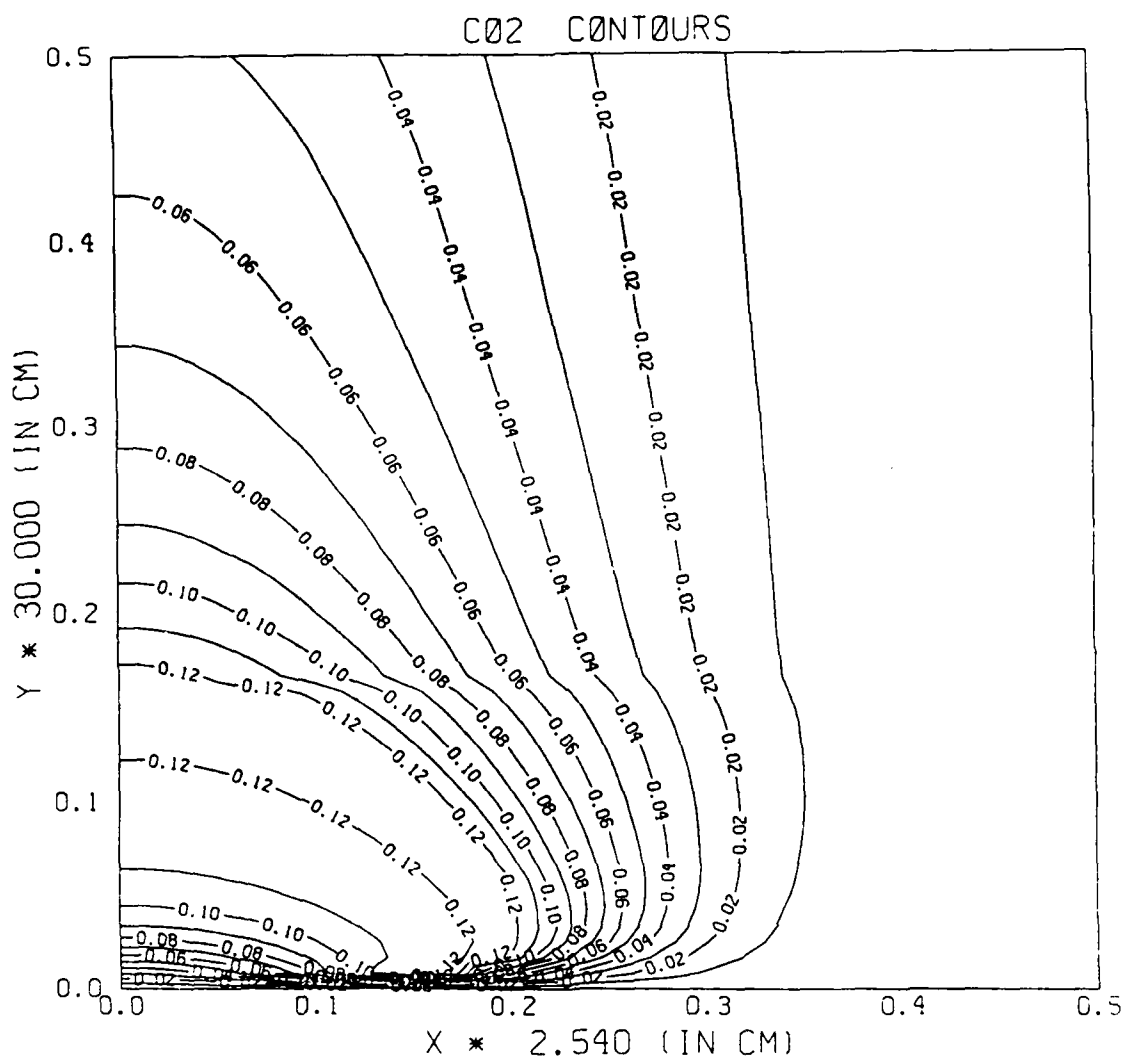


Figure 14: Flame sheet CO_2 (mass fraction) distribution in a two-dimensional axisymmetric diffusion flame.

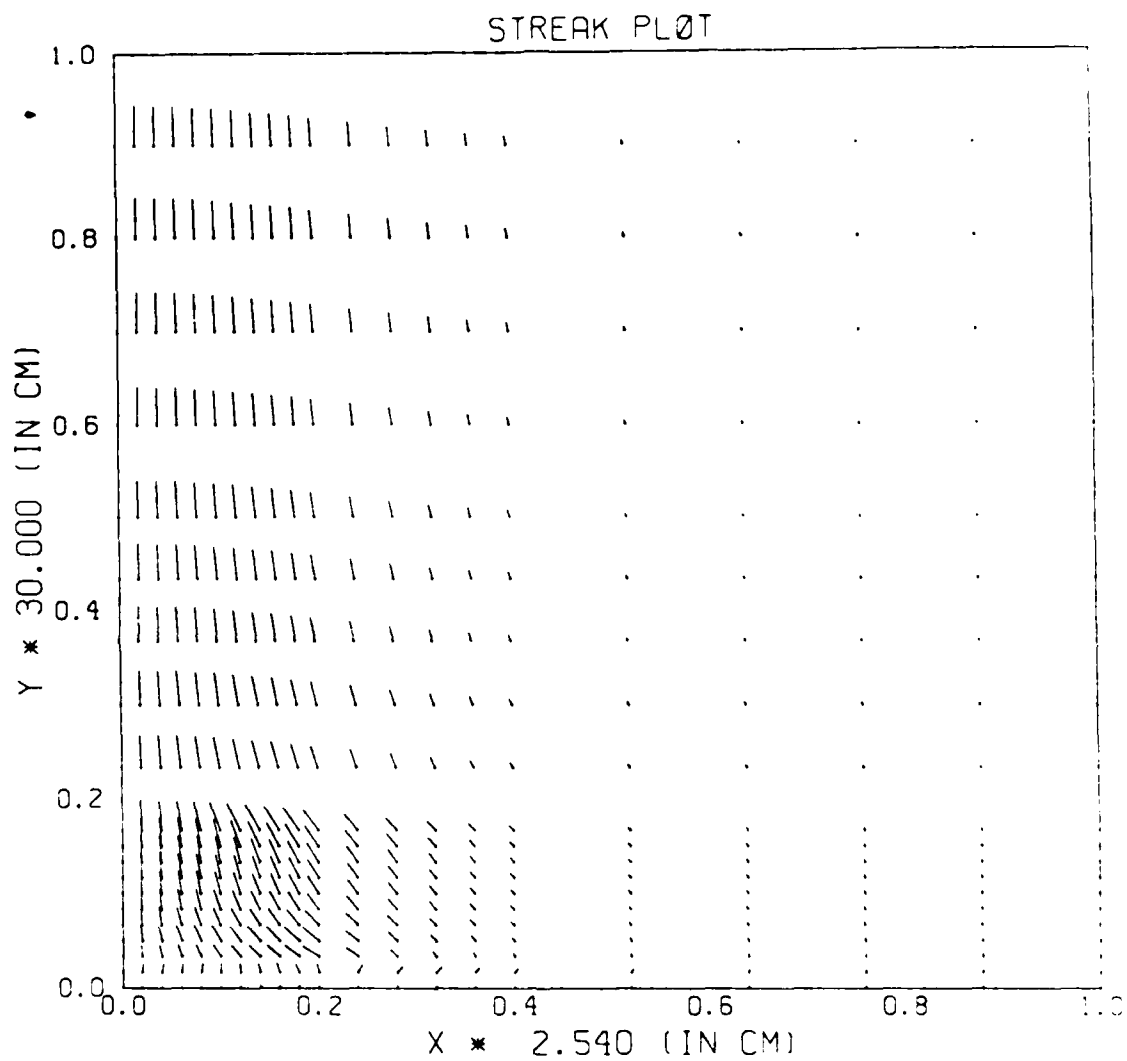


Figure 15: Velocity (vector) distribution in a two-dimensional axisymmetric diffusion flame.

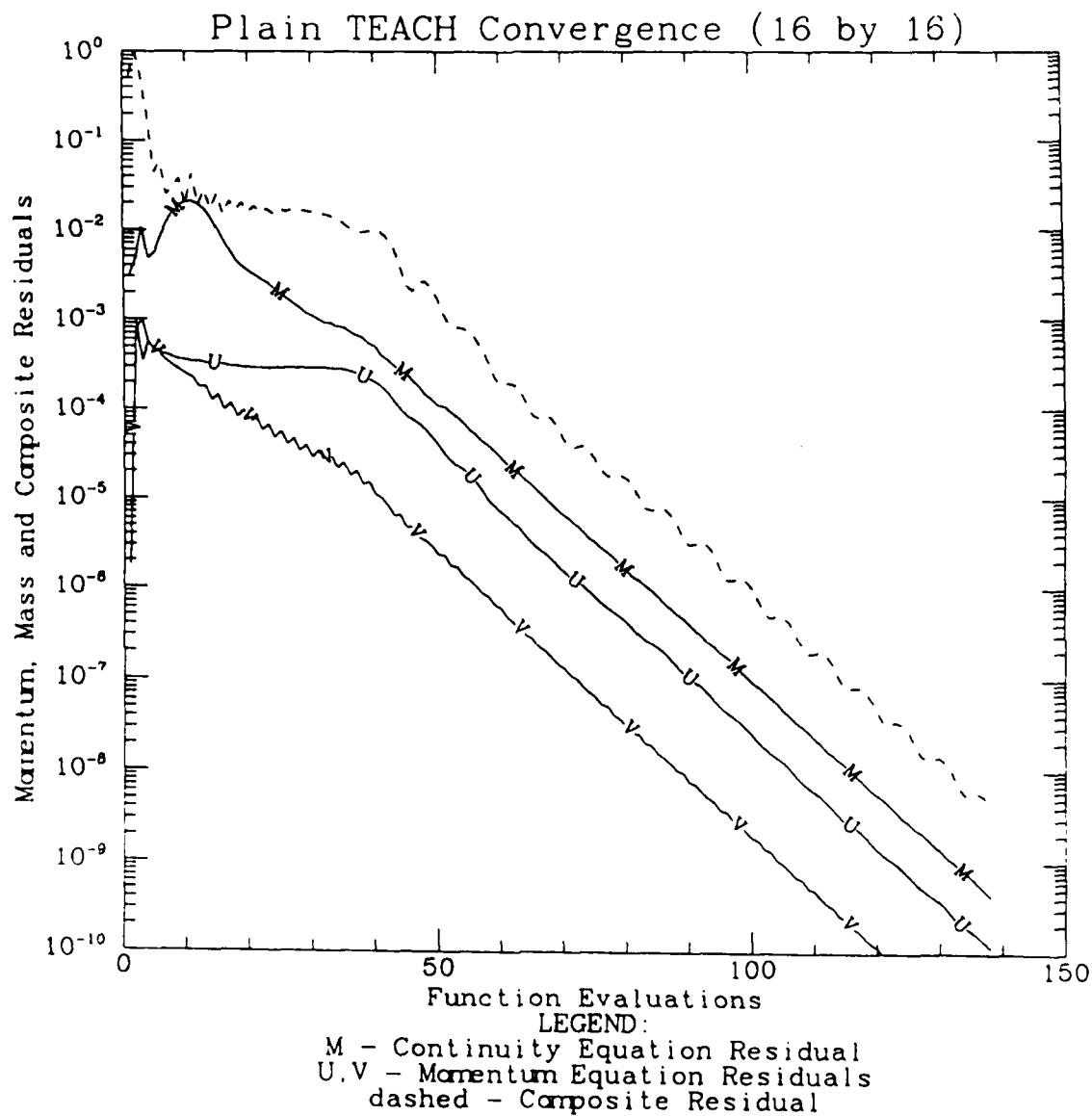


Figure 16: Residuals plotted against number of function evaluations (approximately proportional to CPU time) for Plain *TEACH* for the test problem on a 16×16 grid.

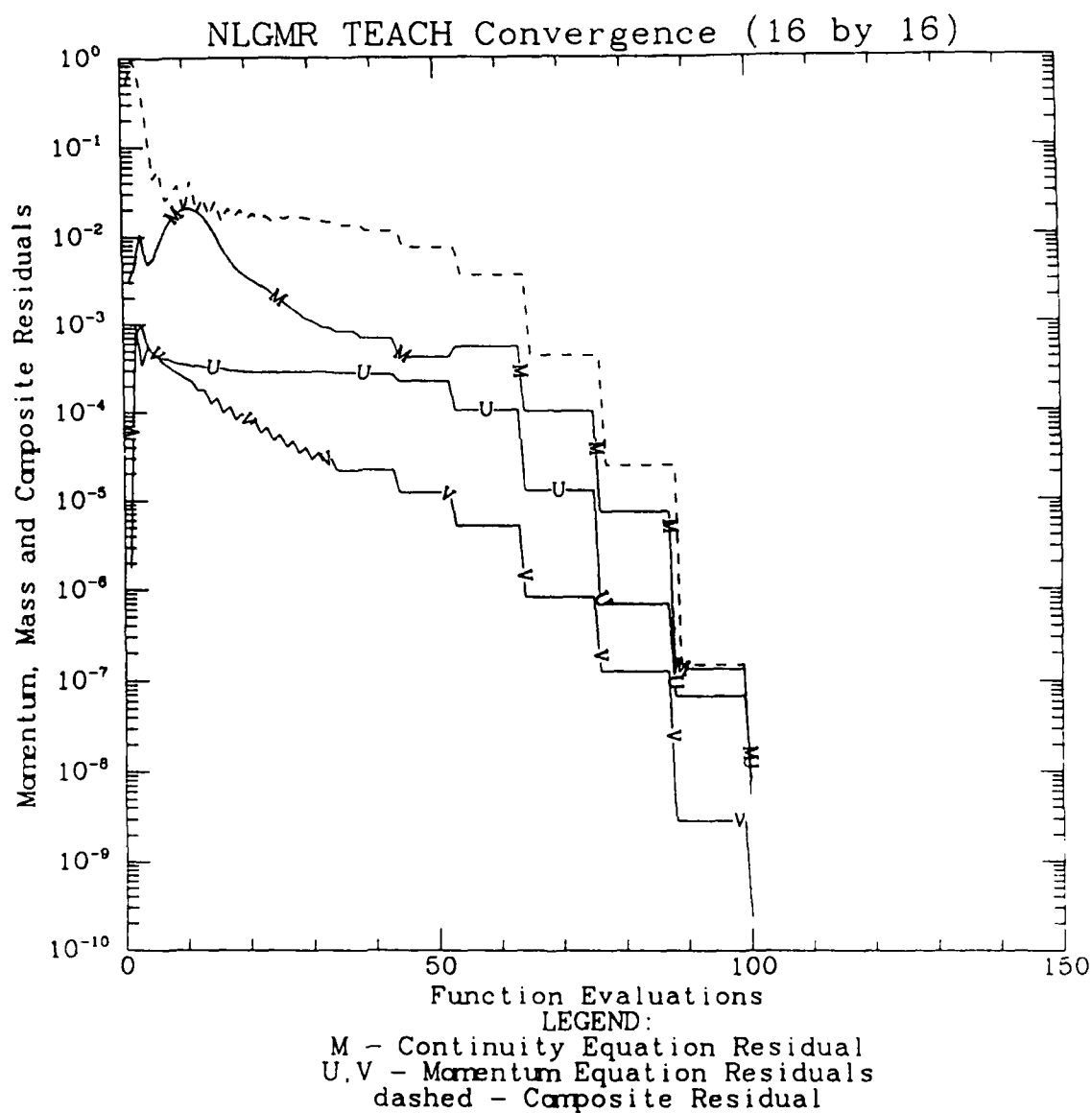


Figure 17: Residuals plotted against number of function evaluations for *NLGMR TEACH* for the test problem on a 16×16 grid.

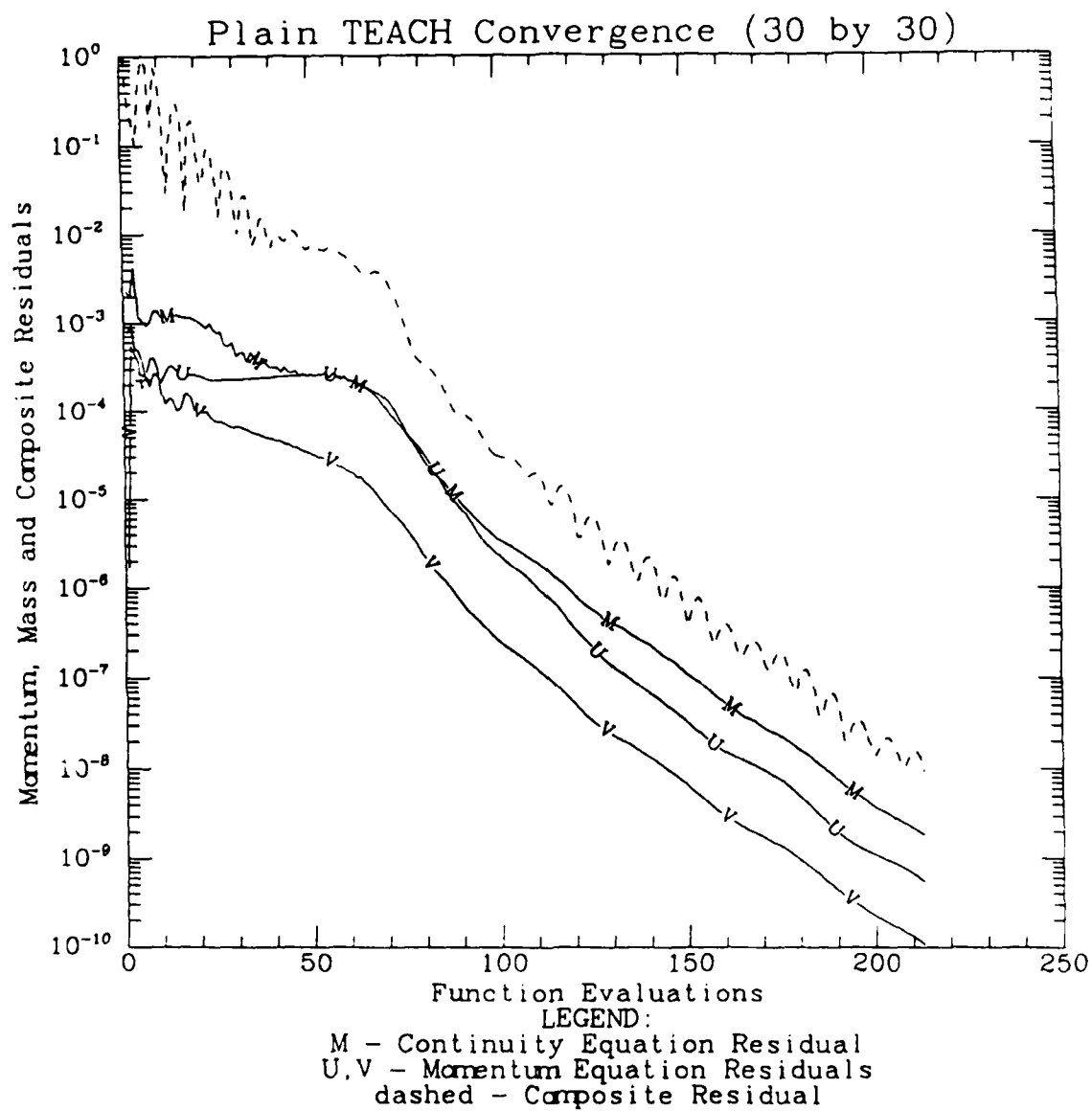


Figure 18: Residuals plotted against number of function evaluations for *TEACH* for the test problem on a 30×30 grid.

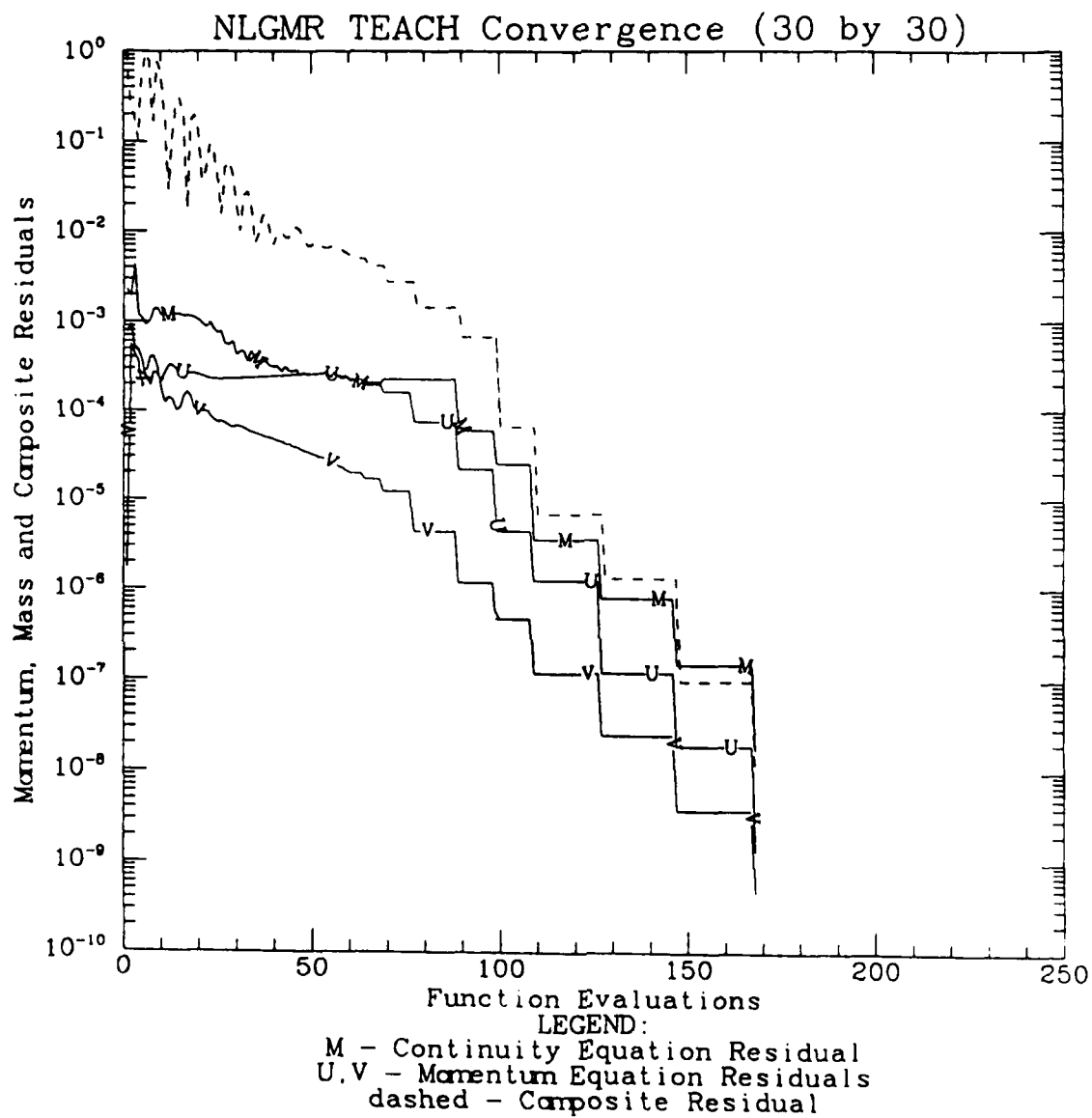


Figure 19: Residuals plotted against number of function evaluations for *NLGMR TEACH* for the test problem on a 30×30 grid.

REFERENCES

1. M. D. Smooke, I. K. Puri and K. Seshadri, *A Comparison Between Numerical Calculations and Experimental Measurements of the Structure of a Counterflow Diffusion Flame Burning Diluted Methane in Diluted Air*, submitted to the Twenty-First Symposium (International) on Combustion, (1986).
2. M. D. Smooke, *Solution of Burner-Stabilized Pre-Mixed Laminar Flames by Boundary Value Methods*, J. Comp. Phys., **48** (1982), p. 72.
3. M. D. Smooke, J. A. Miller and R. J. Kee, *Determination of Adiabatic Flame Speeds by Boundary Value Methods*, Comb. Sci. and Tech., **34** (1983), p. 79.
4. M. D. Smooke, *An Error Estimate for the Modified Newton Method with Applications to the Solution of Nonlinear Two-Point Boundary Value Problems*, J. Opt. Theory and Appl., **39** (1983), p. 489.
5. J. Kautsky and N. K. Nichols, *Equidistributing Meshes with Constraints*, SIAM J. Sci. Stat. Comput., **1** (1980), pp. 499.
6. R. D. Russell, *Mesh Selection Methods*, Proceedings of the Conference for Working Codes for Boundary Value Problems in ODE's, B. Childs et al., eds., Springer-Verlag, New York, (1979).
7. D. B. Spalding, *The Theory of Flame Phenomena with a Chain Reaction*, Phil. Trans. Roy. Soc. London. **249A** (1956), p. 1.
8. D. B. Spalding, D. L. Stephenson and R. G. Taylor, *A Calculation Procedure for the Prediction of Laminar Flame Speeds*, Comb. and Flame., **17** (1971), p. 55.
9. C. K. Westbrook and F. L. Dryer, *A Comprehensive Mechanism for Methanol Oxidation*, Combust. Sci. and Tech., **20** (1979), p. 125.
10. C. K. Westbrook and F. L. Dryer, *Prediction of Laminar Flame Properties of Methanol Air Mixtures*, Comb. and Flame, **37** (1980), p. 171.
11. G. Dixon-Lewis, G. (1967), *Flame Structure and Flame Reaction Kinetics I. Solution of Conservation Equations and Application to Rich Hydrogen-Oxygen Flames*, Proc. Roy. Soc. London. **298A** (1967), p. 495.
12. A. D. Gosman, F. J. K. Ideriah and M. P. Arnal, *TEACH-2E: A General Computer Program for Two-Dimensional, Turbulent, Recirculating Flows*, Technical Report FM-83-2, University of California, Berkeley, (1983).
13. S. V. Patankar, *A Calculation Procedure for Two-Dimensional Elliptic Situations*, Num. Heat Trans., **4** (1981), p. 409.

14. L. H. Back and E. J. Roschke, *Shear-Layer Flow Regimes and Wave Instabilities and Reattachment Lengths Downstream of an Abrupt Circular Channel Expansion*, Trans. ASME, J. Appl. Mech.", **94** (1972), p. 677.
15. T. F. Chan and K. R. Jackson, *Nonlinearly Preconditioned Krylov Subspace Methods for Discrete Newton Algorithms*, Technical Report DCS/RR-259, Yale University, (1983).
16. L. B. Wigton, N. J. Yu and D. P. Young, *GMRES Acceleration of Computational Fluid Dynamics Codes*, Proceedings of the 1985 AIAA Conference, AIAA, Denver, (1985).
17. Y. Saad and M. H. Schultz, *GMRES: A Generalized Minimum Residual Algorithm for Solving Nonsymmetric Linear Systems*, Technical Report DCS/RR-254, Yale University, (1983).
18. H. C. Elman, *Iterative Methods for Large, Sparse, Nonsymmetric Systems of Linear Equations*, Technical Report DCS/RR-229, Yale University, (1982).
19. H. Elman and D. Baxter, *PCGPAK User's Guide*, Technical Report, Scientific Computing Associates, (1984).

END

2-87

DTIC